Master-Thesis MAS-IT 06-02

Schlussbericht

Terminal PMS

Projektbezeichnung	Terminal PMS	
SWS Diplomarbeit	MAS-06.02.16	
Klasse	MAS-06-02	
Diplomanden	Beat Bösiger Niesenblickstrasse 8b 3600 Thun Tel.: +41 33 222 48 77 Fax: +41 33 222 48 78 Mob.: +41 79 643 19 75 E-Mail: beat@studio4master.ch	Heinrich Neiger Silberhornstrasse 10 3600 Thun Mob.: +41 79 661 46 27 E-Mail: heinineiger@hotmail.com
Experte	Joachim Fulde Lutertalstrasse 27 3065 Bolligen Mob.: +41 79 467 89 76 E-Mail: joachim.fulde@bluewin.ch	
Betreuer	Arjan DJ Paijralbe Tel.: +41 33 439 07 92 E-Mail: a.paijralbe@meyerburger.ch	
Erstellt am	18. Februar 2009	
Zuletzt geändert am	19. Februar 2009	
Bearbeitungszustand	In Bearbeitung Vorgelegt X Fertig gestellt	
Dokument	008_Schlussbericht.doc	
Datenablage	Unter Subversion Repository „pms_terminal“ Verzeichnis 010_Trunk\MasterThesis\010_Dokumentation\090_Berichte	

Abstract:
In einem Produktionsprozess von Si-Wafern wird immer mehr nach Rückverfolgbarkeit verlangt.
Die Arbeitsschritte ohne Maschinen werden mit einem Terminal ausgerüstet. Es soll ein Framework zum Erstellen von Terminalanwendungen inkl. einer Referenzanwendung und zwei Hardware-Adaptern erstellt werden.

Management Summary

In dieser Arbeit geht es darum ein Frameworks zu entwickeln, welches als Basis für die Erstellung einer Terminalanwendung benutzt werden kann. Um die Funktion des Frameworks zu Testen, soll zusätzlich eine Referenzanwendung des Brick Separieres implementiert werden. Damit Ein- resp. Ausgaben getätigt werden können, sollen zusätzlich zwei Hardware-Adapter entwickelt werden. Der eine Hardware-Adapter ist für einen Barcode-Leser und der Andere für einen Label-Drucker.

Das Terminal ist ein Teilprojekt des Gesamtsystems PMS. Während dieser Arbeit ist kein funktionsfähiges PMS, resp. Integration-Service (IS) vorhanden, dies aufgrund einer Verspätung des Gesamtprojekts. Der IS dient als Basis für die Kommunikation mit dem PMS. Wegen dieser Verspätung wurde entschieden, dass in dieser Arbeit selber ein IS entwickelt wird, der als Simulator des PMS dient. Das bedeutet, dass die Hardware-Adapter statt an den IS, direkt ins Framework implementiert wurde.

Es konnte ein funktionsfähiges Framework mit integrierten Hardware-Adaptern erstellt werden. Das Framework muss für den effektiven Einsatz in der Industrie an das PMS angepasst werden. Zusätzlich entstanden während der Arbeit zusätzliche Anforderungen, welche aufgrund des zu hohen Arbeitsaufwandes nicht in das Projekt einfliessen konnte. Der Hardware-Adapter für den Barcode-Leser konnte vollständig umgesetzt werden. Der Hardware-Adapter für den Label-Drucker wurde nur simuliert, da der Drucker zu spät evaluiert und beschafft wurde. Die Referenzanwendung Brick Separierer konnte fertig gestellt werden, da der Prozessablauf jedoch nicht endgültig definiert ist, wurde ein provisorischer Prozessablauf erstellt und erstellt.

Es wird empfohlen, das Framework anhand der zusätzlichen Anforderungen zu erweitern und einzusetzen. Falls der Prozessablauf nicht stark ändert, kann der Brick Separierer mit dem derzeitigen Stand eingesetzt werden.

Die Ziele im Pflichtenheft wurden hoch gesteckt, konnten jedoch vollständig umgesetzt werden.

Bis auf die Abnahmen konnten die Termine gehalten werden. Die Abnahmen verzögerten sich, aufgrund des fehlenden Label-Druckers.

Es entstanden keine unvorhergesehene Kosten.

Inhaltsverzeichnis

1 Einleitung 7
1.1 Übersicht 7
1.2 Ausgangslage 7
1.3 Zweck und Ziele 7
1.4 Produktumfang 8
1.5 Definitionen, Akronyme, Abkürzungen 8
1.5.1 Projektspezifisches Glossar 8
1.5.2 Anforderungsspezifisches Glossar/Sprachnormalisierung 9
1.6 Referenzen 10
1.7 Konkurrenzprodukte 10
2 Aufgabenstellung 11
2.1 Hardware-Adapter 11
2.1.1 Label-Drucker 11
2.1.2 Barcode-Leser 11
2.2 Framework 11
2.3 Referenz-Anwendung “Brick Separierer” 12
3 Organisation 12
3.1 Projektplanung 12
3.2 Eingesetzte Werkzeuge 12
4 Analyse und Synthese 13
4.1 Übersicht Gesamtfunktionalität 13
4.2 Komponentenübersicht 14
4.3 Hardware-Adapter 15
4.3.1 Label-Drucker 15
4.3.2 Barcode-Leser 19
4.4 Framework 22
4.4.1 Anwendungsfallmodell 22
4.4.2 Beschreibungen der Anwendungsfälle Entwickler 26
4.4.3 Beschreibungen der Anwendungsfälle System 26
4.4.4 Beschreibungen der Anwendungsfälle Operator 26
4.4.5 Synthese 26
4.4.6 Fachklassendiagramm 30
4.4.7 Beschreibung Fachklassendiagramm 31
4.5 Referenz-Anwendung „Brick Separierer“ 32
4.5.1 Anwendungsfallmodell 32
4.5.2 Beschreibungen der Anwendungsfälle 32
4.5.3 Ablauf 33
5 Prototyping 35
5.1 Hardware-Adapter 35
5.2 Framework 35
5.3 Referenz-Anwendung „Brick Separierer“ 35
6 Design 36
6.1 Hardware-Adapter 36
6.1.1 Label-Drucker 36
6.1.2 Barcode-Leser 36
6.2 Framework 37
6.2.1 Paketaufteilung 37

6.2.2 GUI Architektur ... 39
6.3 Referenz-Anwendung “Brick Separierer” ... 40
6.4 API ... 40
7 Test ... 41
7.1 Hardware-Adapter ... 41
7.1.1 Unit-Test ... 41
7.1.2 Test Szenarien ... 41
7.1.3 Test-Anwendungen ... 41
7.1.4 Abnahme ... 41
7.2 Framework ... 42
7.2.1 Unit-Test ... 42
7.2.2 Test Szenarien ... 42
7.2.3 Test-Anwendung ... 42
7.2.4 Abnahme ... 42
7.3 Referenz-Anwendung “Brick Separierer” ... 43
7.3.1 Unit-Test ... 43
7.3.2 Test ... 43
7.3.3 Abnahme ... 43
7.4 Integrationstest ... 44
7.4.1 Hardware ... 44
7.4.2 Software ... 44
7.4.3 Vorgehen ... 45
8 Resultate ... 46
8.1 Hardware-Adapter ... 46
8.1.1 Abnahme ... 46
8.1.2 Label-Drucker ... 46
8.2 Framework ... 47
8.2.1 Software Update ... 48
8.2.2 Abnahme ... 48
8.3 Referenz-Anwendung “Brick Separierer” ... 49
8.3.1 Abnahme ... 51
8.4 Dokumentation ... 51
8.4.1 “Entwicklerhandbuch” ... 51
8.4.2 API ... 52
8.5 Integration ... 52
8.5.1 Integrationstest ... 52
8.6 Projektplan ... 52
9 Diskussion ... 53
9.1 Projektablauf ... 53
9.2 Framework ... 54
9.2.1 WPF ... 54
9.2.2 Spezifikation ... 54
9.2.3 Integration Service (IS) ... 54
9.2.4 Software-Update ... 54
9.3 Referenz-Anwendung “Brick Separierer” ... 54
10 Ausblick ... 55
10.1 Dokumentation ... 55
10.2 Hardware-Adapter ... 55
10.2.1 Geräte mit Ein-/Ausgabe ... 55

10.2.2 Label-Drucker 55
10.2.3 Barcode-Leser 55
10.3 Framework 56
10.4 Referenz-Anwendung “Brick Separierer” 56
11 Verzeichnisse 57
11.1 Abbildungsverzeichnis 57
11.2 Tabellenverzeichnis 58
12 Anhang 60
12.1 Projektplan 60
12.2 Anwendungsfall-Beschreibungen 62
12.2.1 Hardware-Adapter 62
12.2.2 Framework 66
12.2.3 Referenz-Anwendung „Brick Separierer“ 81
12.3 Test Szenarien 84
12.3.1 Hardware-Adapter 84
12.3.2 Framework 88
12.4 Klassendiagramme 99
12.4.1 Hardware-Adapter 99
12.4.2 Framework 101
12.4.3 Referenz-Anwendung „Brick Separierer“ 109

Dokumentversionen

Tabelle 1: Dokumentversionen

Änderung			Geänderte Kapitel	Beschreibung der Änderung	Autor	Zustand
Nr.	Datum	Version				
1	06.01.2009	0.0.1	Alle	Aufbau gestaltet	Neh	In Bearbeitung
2	30.01.2009	0.0.2	Diskussion	Inhalt eingefügt	Neh	In Bearbeitung
3	16.02.2009	0.0.3	Test Kapitel	Eingefügt	Böb	In Bearbeitung
4	17.02.2009	0.0.4	Test Kapitel Diskussion Kapitel	Ergänzt	Böb	In Bearbeitung
5	18.02.2009	0.0.5	Alle	Abgefüllt mit Inhalt	Böb	In Bearbeitung
6	19.02.2009	0.1.0	Alle	Abgefüllt mit Inhalt	Böb/ Neh	In Bearbeitung
7	19.02.2009	1.0.0	Alles	Fertiggestellt	Böb/ Neh	Fertiggestellt

1 Einleitung

1.1 Übersicht

Dieses Dokument ist so aufgebaut, dass jedes Kapitel jeweils in die drei Themen Hardware-Adapter, Framework und die Referenzanwendung „Brick Separierer“ unterteilt ist.

1.2 Ausgangslage

Die Produktion von Silizium Wafern für Solarzellen erfolgt über mehrere Prozessstufen. Hier setzt das von Meyer Burger entwickelte Prozess Management System (PMS) an. In der Abbildung 1 ist ersichtlich, wie ein Produktionsprozess aussehen kann. Nach dem Giessen der Ingots finden Messungen statt. Anschliessend wird der Ingot in die gewünschte Form geschnitten. Nach diversen Vorgängen, wie Schleifen und Aufkleben wird der Ingot mit einer Drahtsäge in Wafer geschnitten, welche darauf hin separiert und gereinigt werden. Am Ende werden die Wafer vermessen.

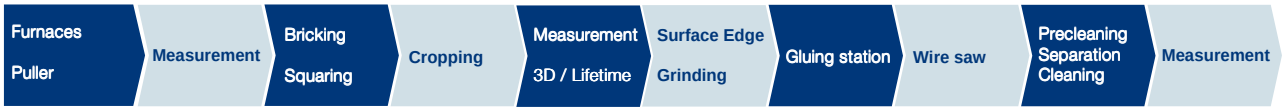


Abbildung 1: Gesamtprozessablauf

Das PMS sammelt alle in der Produktion anfallenden Produkt- und Prozessinformationen, bildet daraus ein abstraktes Produktionsabbild und ermittelt für jede Prozessstufe das Laufzeitverhalten.

Jeder Anlageteil, welcher an das Meyer Burger Leitsystem (PMS) angebunden wird, enthält einen Integrations-Service.

Der Integrations-Service ist eine Software mit der Aufgabe, den Datenfluss der Anlageteile zu koordinieren. Die Daten werden mittels Netzwerk an das Leitsystem (PMS) gesendet und dort zentralisiert.

Für manuelle oder teilautomatisierte Arbeitsstationen wird ein Terminal eingesetzt, über das die notwendigen oder ergänzenden Eingaben möglich sind. Je nach Prozessstufe kann das Terminal über verschiedene Peripheriegeräte (Scanner für die automatische Identifikation / Label-Drucker zur Beschriftung des Materials) verfügen.

1.3 Zweck und Ziele

Anhand dieses Dokumentes sind die Resultate und Erkenntnisse für die aus der Master Thesis entstandene „Terminal PMS“-Software ersichtlich.

1.4 Produktumfang

Das Produkt Terminal PMS beinhaltet drei Teile:

- Ein Framework zum Erstellen von Terminal-Anwendungen.
- Zwei Hardware-Adapter für den Label-Drucker und den Barcode Leser
- Eine Referenzanwendung „Brick Separierer“

1.5 Definitionen, Akronyme, Abkürzungen

1.5.1 Projektspezifisches Glossar

Tabelle 2: Projektspezifisches Glossar

Vokabel	Bedeutung/Verwendung
Bilder	Dateien mit grafischem Bildinhalt, wie Fotos
Brick	Ein Brick ist ein quadrierter Siliziumblock. Dieser Brick wird im Prozessschritt «Wafer Trennen» in dünne Scheiben von 140 --- 200 Mikrometer aufgetrennt.
Brick-Separierer	Ein Brick-Separier ist eine Arbeitsstation, wo ein Terminal eingesetzt wird. Das ist die Referenzanwendung in dieser Diplomarbeit. Beim Brick-Separierer werden die einzelnen Bricks grob gereinigt, auf Qualität geprüft und mit einer Etikette versehen, um die Bricks anschliessend an die nächste Arbeitsstation weiterzugeben.
GUI	Graphical User Interface (Graphische Bediener Schnittstelle)
IS	Integrations-Service, das ist eine Software, welche als Schnittstelle zum PMS dient
Konfigurationsdatei	In der Konfigurationsdatei für eine Terminal-Anwendung werden alle variablen Einstellungen der Anwendung abgelegt. Jede Terminal-Anwendung verfügt über eine solche Konfigurationsdatei. Die Hardware-Adapter verfügen zusätzlich über eigene Konfigurationsdateien.
PMS	Prozess Management System (Meyer Burger Leitsystem)
Produktinformationen	Das sind Eigenschaften, resp. Informationen zu einem Produkt (Ingot/Brick).
Prozessinformationen	Das sind Informationen zum Prozess, wie Zeiten oder anderen Prozessparametern
SEMI E10	SEMI E10 ist ein Standard zur Erfassung der Maschinenbetriebszeiten und Maschinen-/Produktionszustände.
Terminal	Bei Arbeitsplätzen, die über keine entsprechende Maschine mit Leitsystem-Anbindung verfügen, ist ein Rechner mit einer speziellen Applikation zur Daten Ein / Ausgabe verfügbar.
Wafer	Kreisrunde oder quadratische Scheibe, die das Substrat (Grundplatte) darstellt, auf dem elektronische Bauelemente (IC, „Chip“) oder photoelektrische Beschichtungen durch verschiedene technische Verfahren hergestellt werden

1.5.2 Anforderungsspezifisches Glossar/Sprachnormalisierung

Tabelle 3: Anforderungsspezifisches Glossar

Vokabel	Bedeutung/Verwendung
Aktivieren	Das entsprechende Element wird so konfiguriert, dass es verfügbar/sichtbar ist.
Anzeigen	Auf dem Bildschirm ausgeben/anzeigen.
Auswählen	Der Benutzer wählt Eine, der möglichen Optionen/Einstellungen mit der Tastatur, Maus oder dem TouchScreen aus.
Deaktivieren	Das entsprechende Element ist nicht verfügbar/sichtbar.
Einfügen	Bei einer Anwendung: Das System fügt Daten ein, die gespeichert werden (siehe Speichern).
Eingeben	Der Benutzer gibt Daten über die physikalische Tastatur oder über eine virtuelle Tastatur auf dem Touch-Screen Monitor ein.
Einlesen	Daten werden über ein Lesegerät eingelesen und gespeichert.
Entwickler	Diejenigen Personen, die Terminal-Anwendungen erstellen.
Hinzufügen	Beim Entwickeln einer Anwendung: ein Element der Anwendung hinzufügen.
Konfigurieren	Die Daten in der Konfigurationsdatei werden gelesen und beim Zielobjekt gesetzt/gespeichert.
Öffnen	Ein Fenster auf dem Bildschirm öffnen.
Speichern	Bei einem Wert: Das System speichert den Wert im Arbeitsspeicher. Bei einer Datei: Das System speichert die Datei auf dem entsprechenden Speichermedium.
System	Das System ist eine Terminal-Anwendung, die mittels des Frameworks erstellt worden ist.

1.6 Referenzen

Die folgende Tabelle gibt einen Überblick, woher die Terminal PMS spezifischen Informationen stammen:

Tabelle 4: Projektspezifische Referenzen

Name des Dokuments	Revision	Dokumentenablage	Kommentar
MeyerBurger PMS.eap	3425	Subversion Repository „pms_design_ea“	Design-Projektdatei für Enterprise Architect
Implementierungsrichtlinien_ PMS.doc	359	Subversion Repository „pms“, Verzeichnis: 010_Projekt_Dok\ Implementierungsrichtlinien	Programmierrichtlinien
Vergabe der ID Nummern.doc	359	Subversion Repository „pms“, Verzeichnis: 010_Projekt_Dok\Pflichtenhefte	Beschreibung Vergabe von ID Nummern
e1000.pdf	359	Subversion Repository „pms“, Verzeichnis: 010_Projekt_Dok\Normen	SEMI E10 Standard
Pflichtenheft PMS-Terminal	104	Subversion Repository „pms_terminal“, Verzeichnis 010_Trunk\MasterThesis\010_Dokumentation\040_Pflichtenheft	Pflichtenheft für das PMS Terminal

1.7 Konkurrenzprodukte

Beim Gesamtprojekt PMS handelt es sich um eine eigene Neuentwicklung der Meyer Burger AG. Das Terminal PMS ist dabei ein Teilprojekt des PMS, das den Integration-Service als weiteres Teilprojekt beinhaltet. Aus dieser Situation heraus existieren keine direkten Konkurrenzprodukte.

2 Aufgabenstellung

Diese Master Thesis beinhaltet drei Themen, welche zusammen den Hauptteil des Terminal PMS bildet. Diese drei Themen sind die Hardware-Adapter Barcode-Leser und Label-Drucker, das Framework und die Referenzanwendung „Brick Separierer“.

In den folgenden Kapitel wir auf diese Themen genauer eingegangen.

2.1 Hardware-Adapter

An das Terminal PMS können diverse Hardware-Adapter, resp. zusätzliche Hardware angeschlossen werden. In dieser Arbeit wurde der Hardware-Adapter für einen Label-Drucker und einen Barcode-Leser implementiert.

2.1.1 Label-Drucker

Der Label-Drucker soll an das Terminal PMS angeschlossen werden. Anhand der Konfiguration im Framework wird der Label-Drucker Adapter dynamisch in der Anwendung eingebunden und die Schnittstelle richtig konfiguriert. Während des Betriebs können auf dem Label-Drucker Barcode-Etiketten anhand einer Vorlage, welche vom PMS vorgegeben ist, ausgedruckt werden.

2.1.2 Barcode-Leser

Der Barcode-Leser soll an das Terminal PMS angeschlossen werden. Anhand der Konfiguration im Framework wird der Barcode-Leser Adapter dynamisch in der Anwendung eingebunden und die Schnittstelle richtig konfiguriert. Während des Betriebs kann der Barcode-Leser Barcodes einlesen und die Daten ans Framework, bzw. an die Terminal-Anwendung weitergeben.

Dies gilt für den folgenden Typ: Powerscan D8330 von Datalogic

2.2 Framework

Das Framework ist die Basis jeder Terminal-Anwendung. Es beinhaltet die Grundfunktionalität:

- Adapter-Loader und Event-Support für Hardware-Adapter.
- Grundaufbau der Visualisierung
- Hilfsmittel (Rechner / Kalender / Bild-Assistent / Testbedingungshilfe / Bildschirmreiniger)
- Basis für die Konfiguration
- Kommunikation mit dem IS (Integration Service)
- Sprachabhängige Unterstützung

Das Ziel des Frameworks ist, dass weitere Terminalanwendungen mit möglichst wenig Aufwand erstellt werden können.

2.3 Referenz-Anwendung “Brick Separierer”

Um eine Umsetzung einer Terminal Anwendung zu zeigen, wird mit dem erstellten Framework eine Referenz-Anwendung realisiert. Diese Referenz-Anwendung wird Brick Separierer genannt, da das Terminal in einem Prozess eingesetzt wird, wo die Bricks separiert werden. Das bedeutet, dass nach dem Sägen mehrere Bricks beim Terminal ankommen, wo ein Mitarbeiter mit Hilfe des Terminals die Bricks separiert. Der Mitarbeiter kann mit dem Barcode-Leser oder von Hand eine Auftragsnummer eingeben und erhält die dazugehörigen Auftragsdaten angezeigt. Nun führt das Terminal den Mitarbeiter durch den Prozess inkl. dem Ausdruck der dazugehörigen Beschriftungen. Dabei geht es primär um die Nachverfolgbarkeit der einzelnen Bricks.

3 Organisation

3.1 Projektplanung

Der erstellte Projektplan befindet sich im Anhang im Kapitel 12.1.

3.2 Eingesetzte Werkzeuge

In der Tabelle 5 sind alle eingesetzten Werkzeuge aufgelistet, die zur Erstellung der Diplomarbeit benutzt worden sind.

Tabelle 5: Eingesetzte Werkzeuge

Bezeichnung	Werkzeug	Hersteller	Version
Projektmanagement/Zeitplan	GanttProject	ganttproject.biz	2.0.7
Dokumentation	Office 2003	Microsoft	2003 SP3
PDF	Office Plugin		
Mindmap	FreeMind	freemind.sourceforge.net	0.9.0
Design/Diagramme	Enterprise Architect	Sparx Systems	7.0.813
Code/Realisierung	Visual Studio 2008	Microsoft	2008 (9.0)
XML Dokumente	Visual Studio 2008	Microsoft	2008 (9.0)
Grafik/Bilder	GIMP	www.gimp.org	2.4.0
Versionierung	TortoiseSVN	www.tigris.org	1.4.8
API-Dokumentation Generator	Doxygen	www.doxygen.org	1.5.8
Visual Studio Plugin: ReSharper	ReSharper		4.1

4 Analyse und Synthese

4.1 Übersicht Gesamtfunktionalität

Um einen Gesamtüberblick zu bekommen, ist ein Mindmap erstellt worden und ist in der folgenden Abbildung dargestellt. Es zeigt in groben Zügen alle Elemente, die mit dem Produkt zusammenhängen.

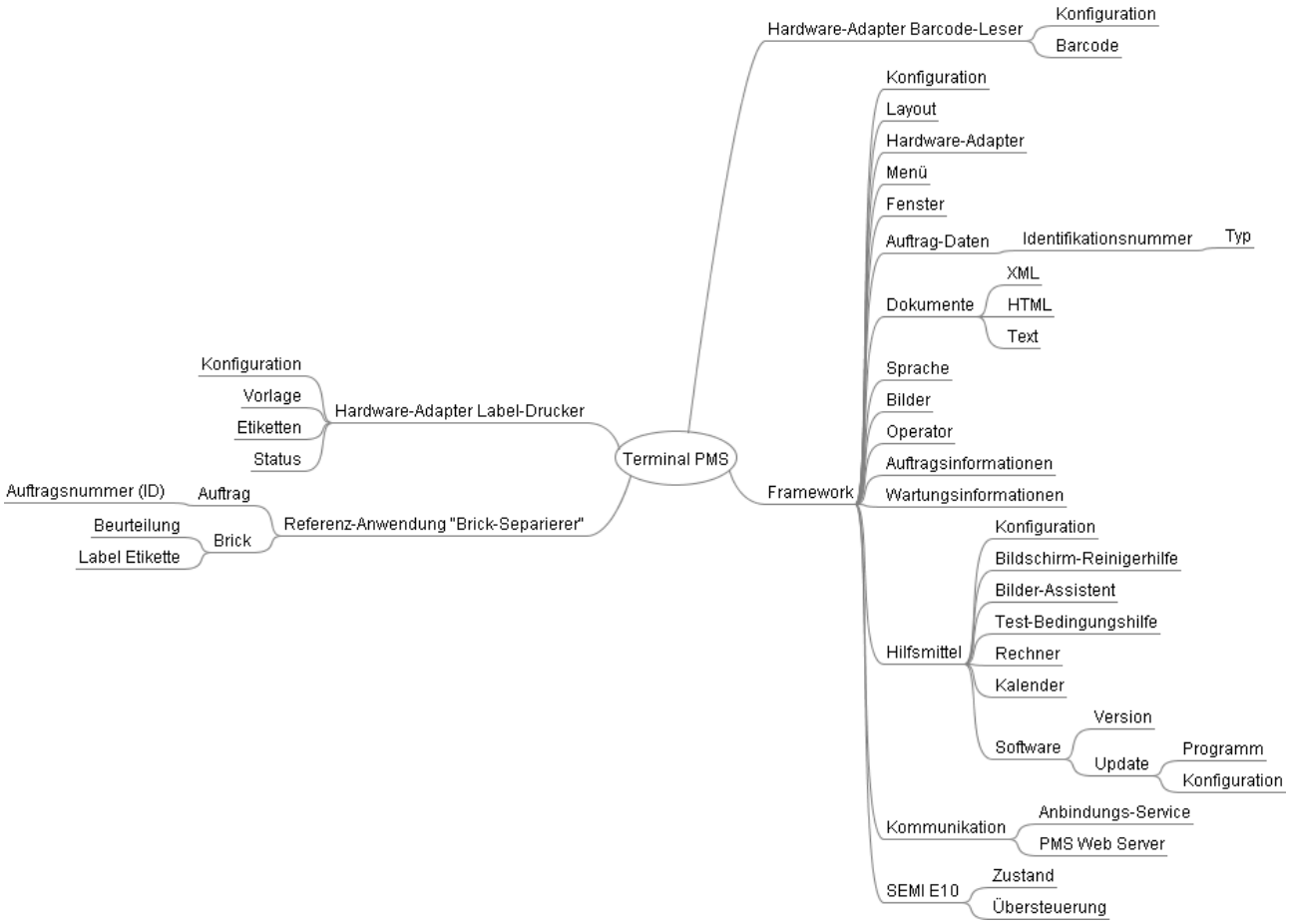


Abbildung 2: Mindmap Gesamtfunktionalität

4.2 Komponentenübersicht

In der folgenden Abbildung 3 sind die zusammenhängenden Komponenten und deren Verknüpfungen ersichtlich.

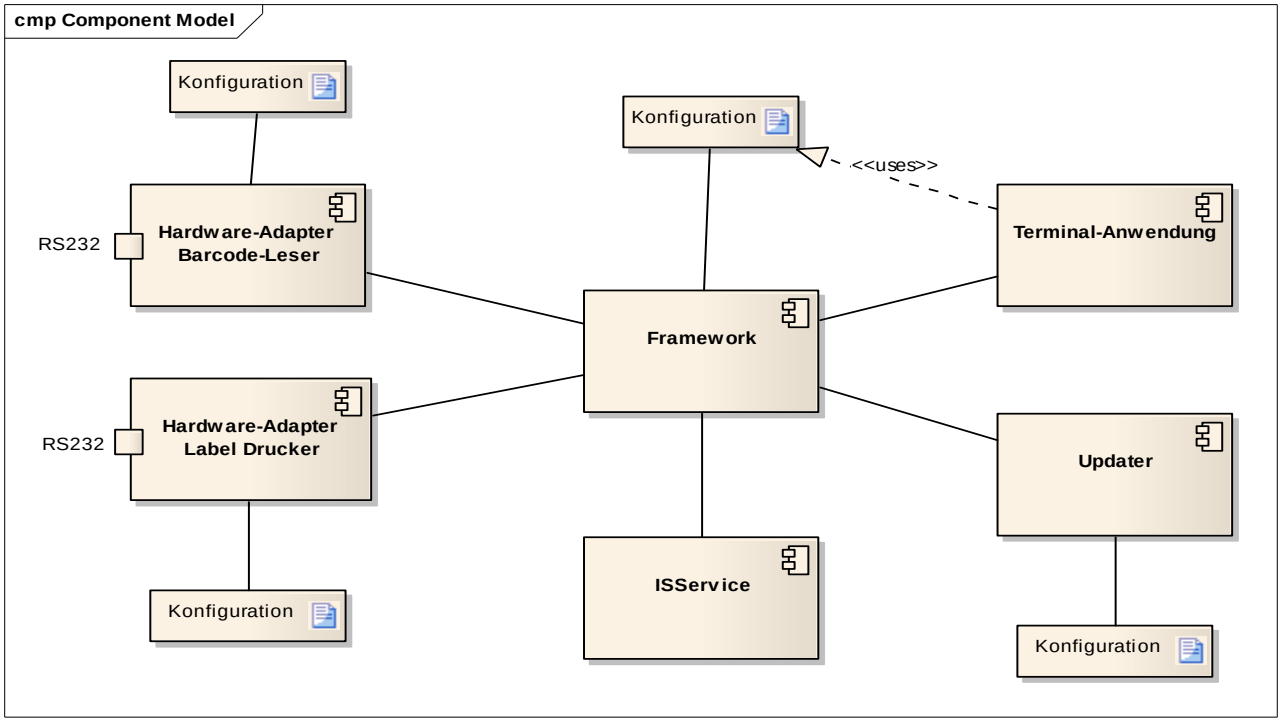


Abbildung 3: Komponentenübersicht

- Hardware-Adapter Label-Drucker: Bibliothek (dll). Der Hardware-Adapter hat eine eigene Konfigurationsdatei zum Konfigurieren der Schnittstelle und zum Konfigurieren des Druckers. Falls in der Konfigurationsdatei der Applikation der Label-Drucker aktiviert wird, wird die Bibliothek dynamisch geladen.

- Hardware-Adapter Barcode-Leser: Bibliothek (dll). Der Hardware-Adapter hat eine eigene Konfigurationsdatei zum Konfigurieren der Schnittstelle. Falls in der Konfigurationsdatei der Applikation der Barcode-Leser aktiviert wird, wird die Bibliothek dynamisch geladen.

- Framework: Bibliothek (dll). Das Herzstück der Terminal-Anwendung. Es beinhaltet die Basisfunktionalität. Es benutzt die von der Applikation übergebene Konfigurationsdatei für alle konfigurierbaren Elemente.

- Terminal-Anwendung: Applikation (exe), sie beinhaltet eigene Fenster und Funktionalitäten. Für die Entwicklung einer Terminal-Anwendung muss die Framework-Bibliothek ins Projekt eingebunden werden.

- ISService: Bibliothek (dll). Sie beinhaltet die Definition der Kommunikationsschnittstelle zum Integration Service.

- Updater: Konsolenanwendung (exe). Ist zuständig für ein Software Update der Terminalanwendung. Beim Aufruf des Updater's beendet der Updater die Terminal-Anwendung, installiert die heruntergeladenenen Dateien und startet die Terminal-Anwendung erneut. Der Updater hat eine eigene Konfigurationsdatei.

4.3 Hardware-Adapter

4.3.1 Label-Drucker

4.3.1.1 Anwendungsfallmodell

In der Abbildung 4 ist das Anwendungsfallmodell für den Label-Drucker abgebildet. Es ist ersichtlich, dass vom PMS eine Vorlage heruntergeladen wird. Dies wird abgespeichert und zum Ausdrucken der Etiketten benutzt. Für die Konfiguration sind zwei Konfigurationsdateien vorgesehen. Die Eine für die Schnittstelle und die Andere für die Konfiguration des Druckers selbst.

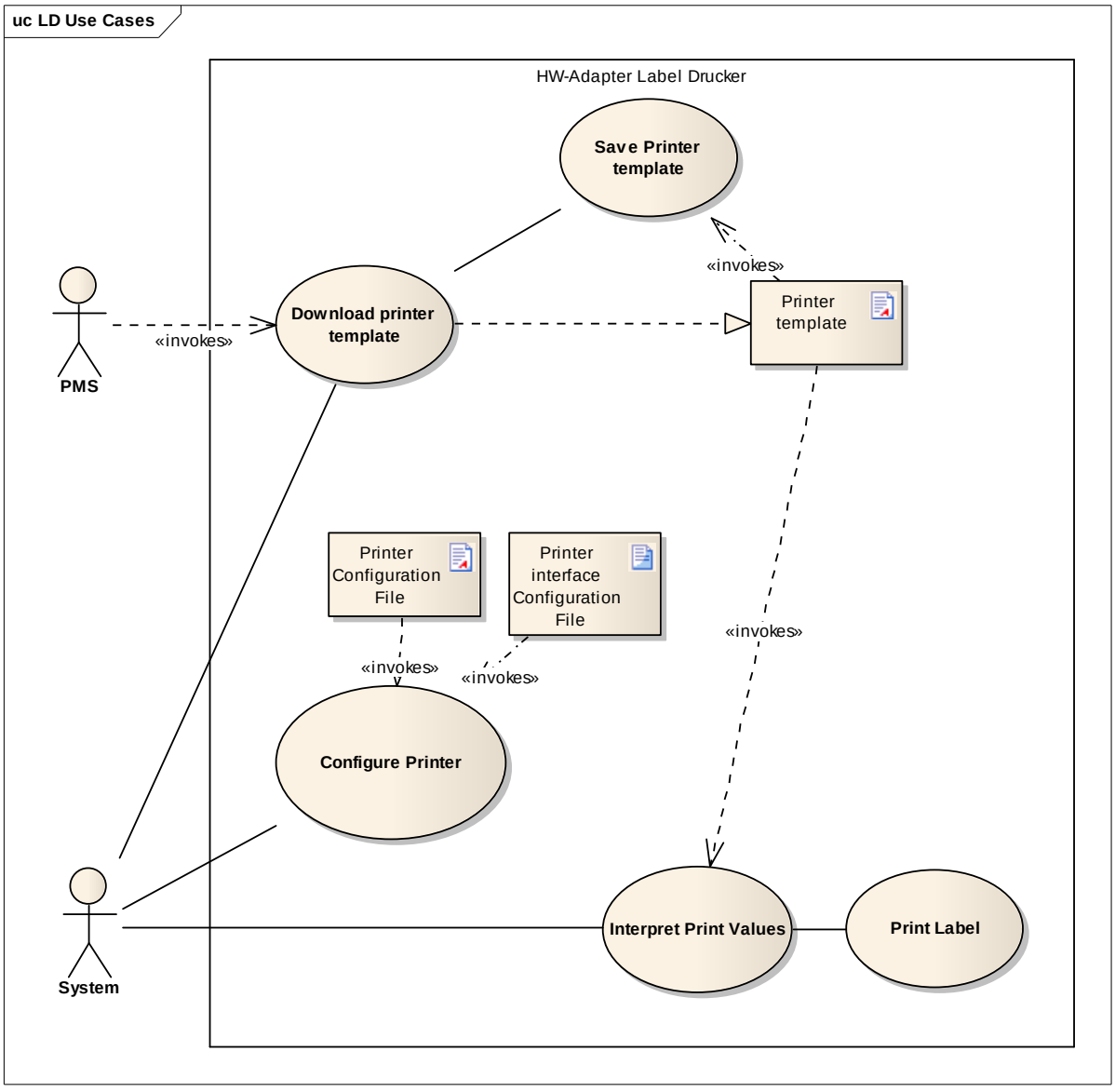


Abbildung 4: Anwendungsfallmodell für den Label-Drucker

4.3.1.2 Beschreibungen der Anwendungsfälle

Aus den Tabellen im Anhang unter Kapitel 12.2.1.1 sind die Beschreibungen der Anwendungsfälle ersichtlich.

4.3.1.3 Synthese

Die folgende Tabelle 6 zeigt das Ergebnis der Zusammentragung von allen Informationen aus Lastenheft, Pflichtenheft, sonstigen Informationsquellen zum Label-Drucker, geordnet in die verschiedenen Elemente, um daraus das Fachklassendiagramm zu erstellen.

Tabelle 6: Synthese Label-Drucker

1. Klassen/Objekte	2. Attribute	3. Operationen
Adapter		Adapter hinzufügen Adapter entfernen Wert_abfragen Konfigurationsstatus weiterleiten Label_drucken
Schnittstelle	Druckinhalt	Schnittstelle öffnen Druckinhalt schreiben Schnittstelle schliessen Druckinhalt erzeugen
Konfiguration	Dateiname Schnittstelle Baudrate Parität Datenbits Stopbit Status	Schnittstelle konfigurieren Konfigurationsdatei öffnen Parameter lesen Konfigurationsdatei schliessen
Vorlage	URL Dateiname Pfad	herunterladen speichern Escape-Sequenzen interpretieren Durch Werte ersetzen

4.3.1.4 Fachklassendiagramm

Aus der Synthese ist das Fachklassendiagramm, wie in der Abbildung 5 zu sehen ist, entstanden. Das Fachklassendiagramm zeigt die Klassen, die benötigt werden, um die aus dem Anwendungsfallmodel entstandenen Geschäftsanwendungsfälle abdecken zu können. Es entspricht deshalb nicht dem definitiven Klassendiagramm und wird im Detaildesign weiterentwickelt.

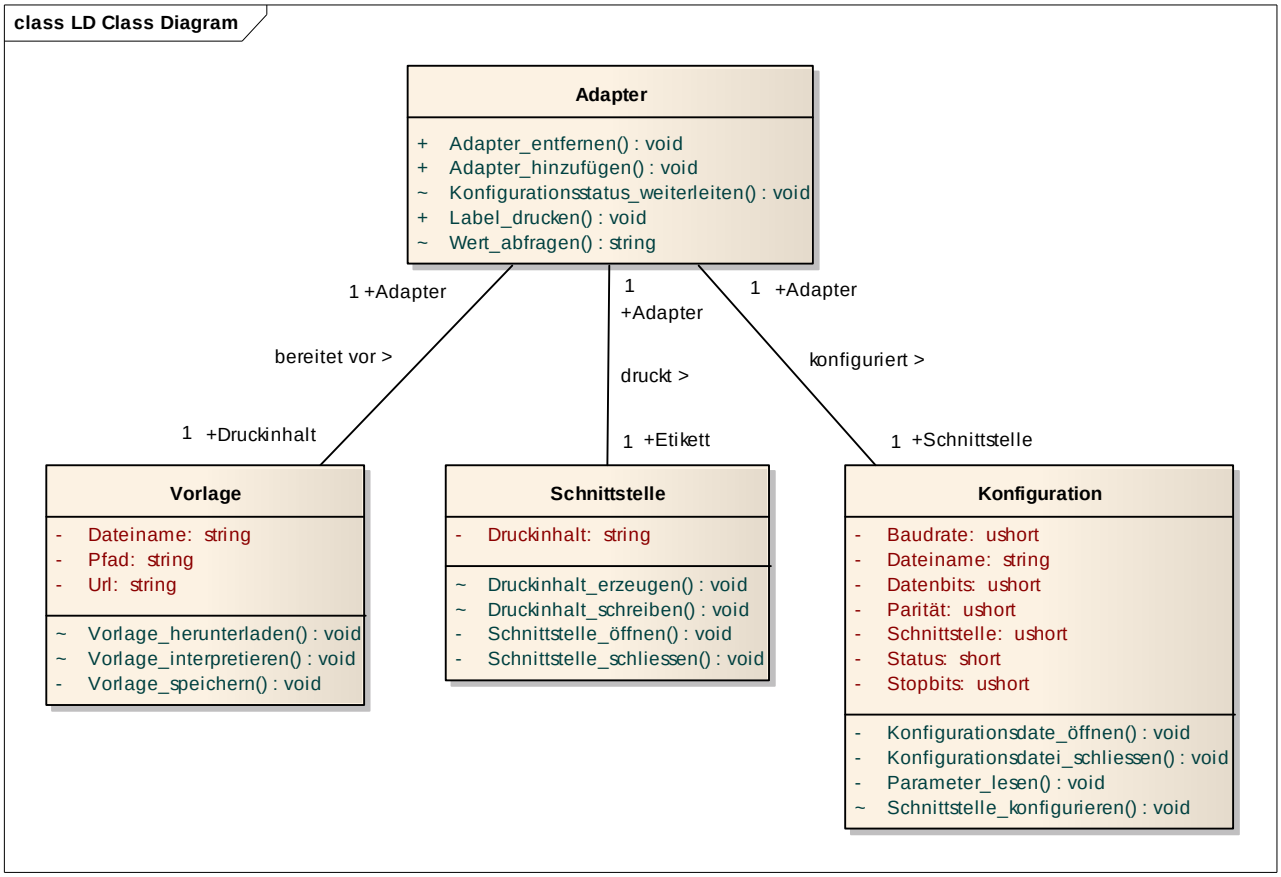


Abbildung 5: Fachklassendiagramm Label-Drucker

4.3.1.5 Beschreibung Fachklassendiagramm

Mit Hilfe der Synthese ist das Fachklassendiagramm in Abbildung 5 entstanden.

Darin enthalten sind die folgenden Klassen:

- Adapter: Die Klasse *Adapter* ist die Hauptklasse des Hardware-Adapter's. Sie bietet die Methoden, um den Adapter zu registrieren und auch wieder zu entfernen. Zusätzlich beinhaltet sie Event und Handler für den Datenaustausch mit dem Framework.

- Konfiguration: Die Klasse *Konfiguration* bietet das Auslesen der Konfigurationselemente aus der entsprechenden Konfigurationsdatei.

- Schnittstelle: Die Klasse *Schnittstelle* stellt die Verbindung zum Gerät über die entsprechende Schnittstelle zur Verfügung. Sie bietet Methoden zum Öffnen und Schliessen der Schnittstelle sowie eine Methode zum Schreiben von Daten zum Gerät.

- Vorlage: Die Klasse *Vorlage* bietet Funktionen zum Herunterladen und Speichern einer vorgegebenen Vorlage sowie den Parser zum Interpretieren des Vorlageninhaltes.

4.3.2 Barcode-Leser

4.3.2.1 Anwendungsfallmodell

Aus dem Anwendungsfallmodell des Barcode-Lesers, gemäss Abbildung 6, erkennt man, dass das System mittels einer Konfigurations-Datei die Schnittstelle zum Barcode-Leser konfiguriert werden kann. Anschliessend ist der Barcode-Leser funktionsfähig.

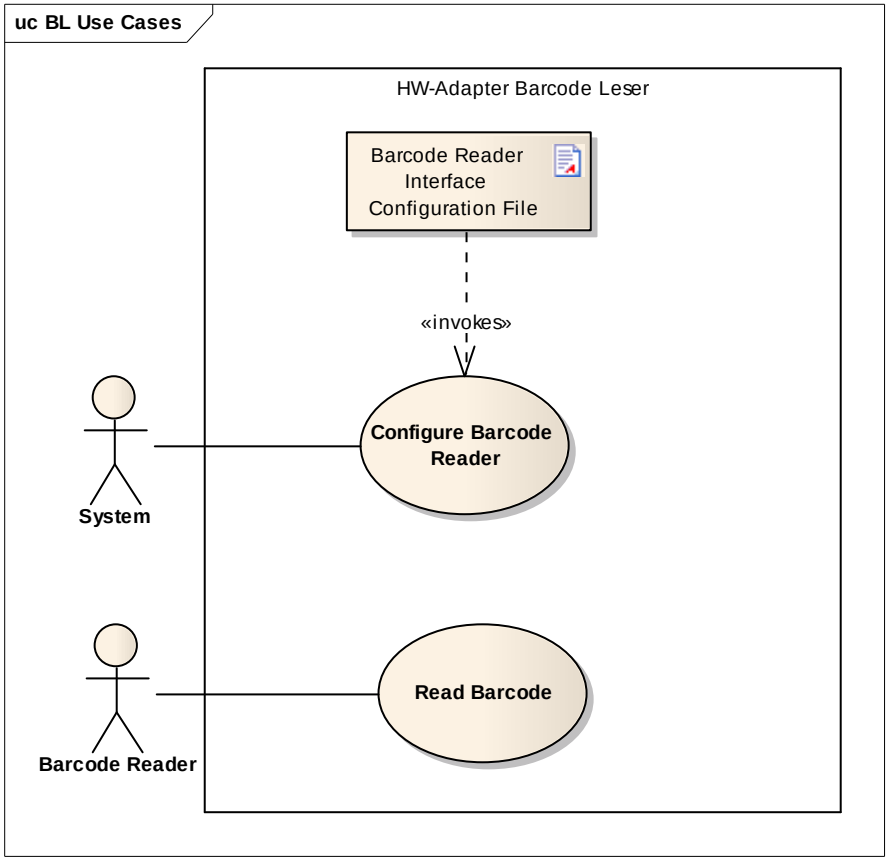


Abbildung 6: Anwendungsfallmodell für den Barcode-Leser

4.3.2.2 Beschreibungen der Anwendungsfälle

Aus den Tabellen im Anhang unter Kapitel 12.2.1.2 sind die Beschreibungen der Anwendungsfälle ersichtlich.

4.3.2.3 Synthese

Die Tabelle 7 zeigt das Ergebnis der Zusammentragung von allen Informationen aus Lastenheft, Pflichtenheft, sonstigen Informationsquellen zum Barcode-Leser, geordnet in die verschiedenen Elemente, um daraus das Fachklassendiagramm zu erstellen.

Tabelle 7: Synthese Barcode-Leser

1. Klassen/Objekte	2. Attribute	3. Operationen
Adapter		Adapter hinzufügen Adapter entfernen Barcode weiterleiten Konfigurationsstatus weiterleiten
Schnittstelle	Barcode	Schnittstelle öffnen Barcode lesen Schnittstelle schliessen
Konfiguration	Dateiname Schnittstelle Baudrate Parität Datenbits Stopbit Status	Schnittstelle konfigurieren Konfigurationsdatei öffnen Parameter lesen Konfigurationsdatei schliessen

4. Multiplizitäten	5. Bedingungen/Periodizitäten	6. Assoziationen
die Schnittstelle der Leser die Konfigurationsdatei ein Barcode Mindestens 1 Parameter den Parametern	Beim Start der Anwendung Schnittstelle konfigurieren	

4.3.2.4 Fachklassendiagramm

Aus der Synthese ist das Fachklassendiagramm, wie in der Abbildung 7 zu sehen ist, entstanden. Das Fachklassendiagramm zeigt die Klassen, die benötigt werden, um die aus dem Anwendungsfallmodel entstandenen Geschäftsanwendungsfälle abdecken zu können. Es entspricht deshalb nicht dem definitiven Klassendiagramm und wird im Detaildesign weiterentwickelt.

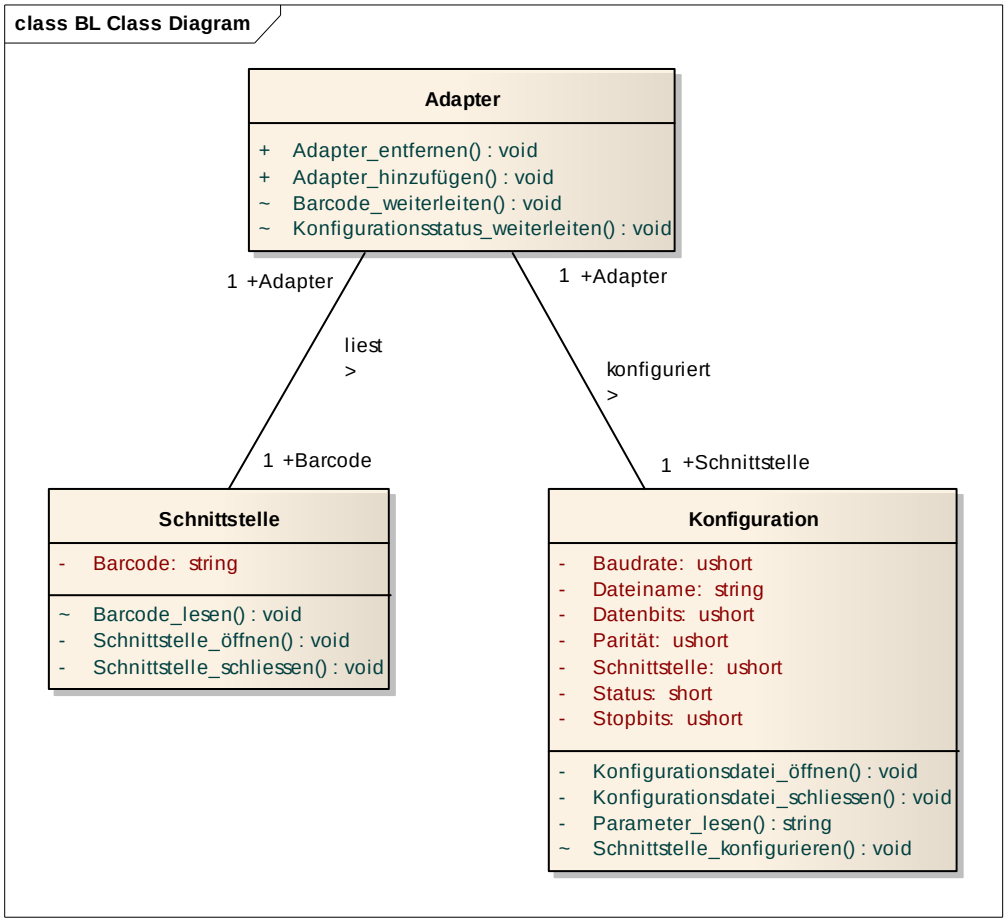


Abbildung 7: Fachklassendiagramm Barcode-Leser

4.3.2.5 Beschreibung Fachklassendiagramm

Mit Hilfe der Synthese ist das Fachklassendiagramm in Abbildung 7 entstanden.

Darin enthalten sind die folgenden Klassen:

- Adapter: Die Klasse *Adapter* ist die Hauptklasse des Hardware-Adapter's. Sie bietet die Methoden, um den Adapter zu registrieren und auch wieder zu entfernen. Zusätzlich beinhaltet sie Event und Handler für den Datenaustausch mit dem Framework.

- Konfiguration: Die Klasse *Konfiguration* bietet das Auslesen der Konfigurationselemente aus der entsprechenden Konfigurationsdatei.

- Schnittstelle: Die Klasse *Schnittstelle* stellt die Verbindung zum Gerät über die entsprechende Schnittstelle zur Verfügung. Sie bietet Methoden zum Öffnen und Schliessen der Schnittstelle sowie eine Methode zum Lesen der Daten vom Gerät.

4.4 Framework

4.4.1 Anwendungsfallmodell

Das folgende Anwendungsfallmodell in Abbildung 8 zeigt drei Akteure:

- Developer: Der Entwickler einer Terminalanwendung. Dem Entwickler müssen die verlangten Grundfunktionalitäten aus dem Pflichtenheft mit dem Framework zur Verfügung stehen. Dazu gehört das Hinzufügen von Hardware-Adaptern, Bildern, Sprachen, das Hinzufügen und Editieren von Fenstern, das Konfigurieren der Anwendung mit dem Aktivieren/Deaktivieren von Identifikationsnummern, Fenstern und Hilfsmittel.

- System: Das System, in diesem Fall die Frameworkumgebung. Es behandelt alle Abläufe und Vorgänge, die mit Kommunikation und Daten im Zusammenhang stehen. Dazu gehört das Laden von konfigurierten Hardware-Adaptern, das Setzen des Produktionsstandes (SemiE10), das Einlesen von Barcodes über den Hardware-Adapter des Barcode-Lesers sowie das Zwischenspeichern von URL's. Zudem muss es in der Lage sein, anhand der entsprechenden URL Dokumente von einem Web-Server herunterzuladen und anzuzeigen.

- PMS: Das Leitsystem bekommt Anfragen von der Applikation und gibt entsprechend Antworten zurück, die in der Applikation weiterverarbeitet werden.

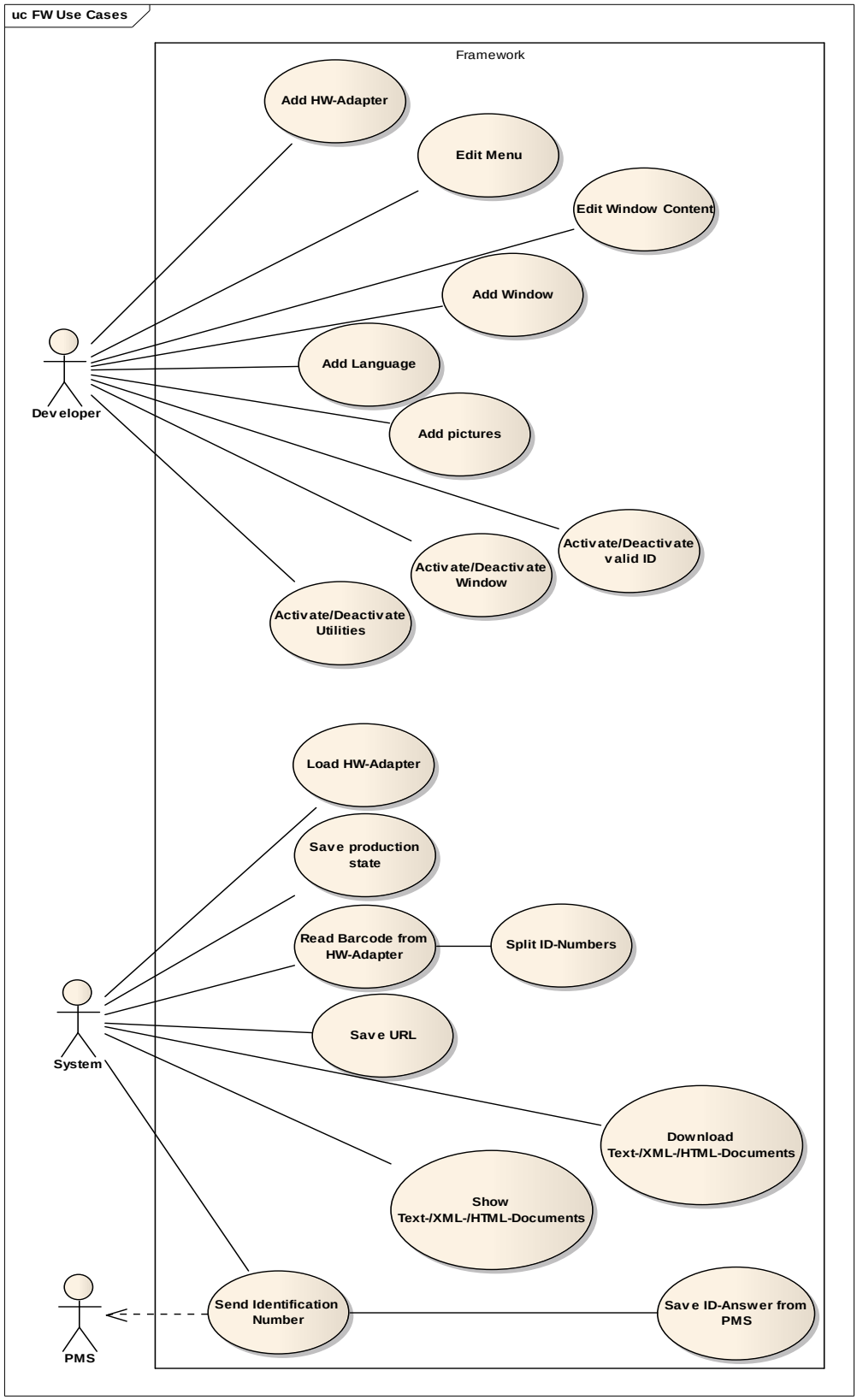


Abbildung 8: Anwendungsfallmodell der Basisfunktionalität im Framework

Das folgende Anwendungsfallmodell in Abbildung 9 zeigt den Operator, der verschiedene Vorgänge ausführen können muss. Dazu gehören folgende Vorgänge:

- Anmelden mit einer gültigen Benutzer-Identifikationsnummer, und das Abbrechen der Anmeldung
- Öffnen und Schliessen von folgenden Fenstern: Arbeitsablauf-Beschreibung, Auftragsbeschreibung, Informationen zum nächsten Auftrag, Wartungsinformationen, Bildschirmreiniger, Bild-Assistent, Testbedingungshilfe, Rechner, Kalender und Software Version. Beim Rechner müssen zudem Berechnungen möglich sein.
- Ändern der Sprache
- Installieren von Software-Updates
- Übersteuern des Produktionsstandes (SemiE10)

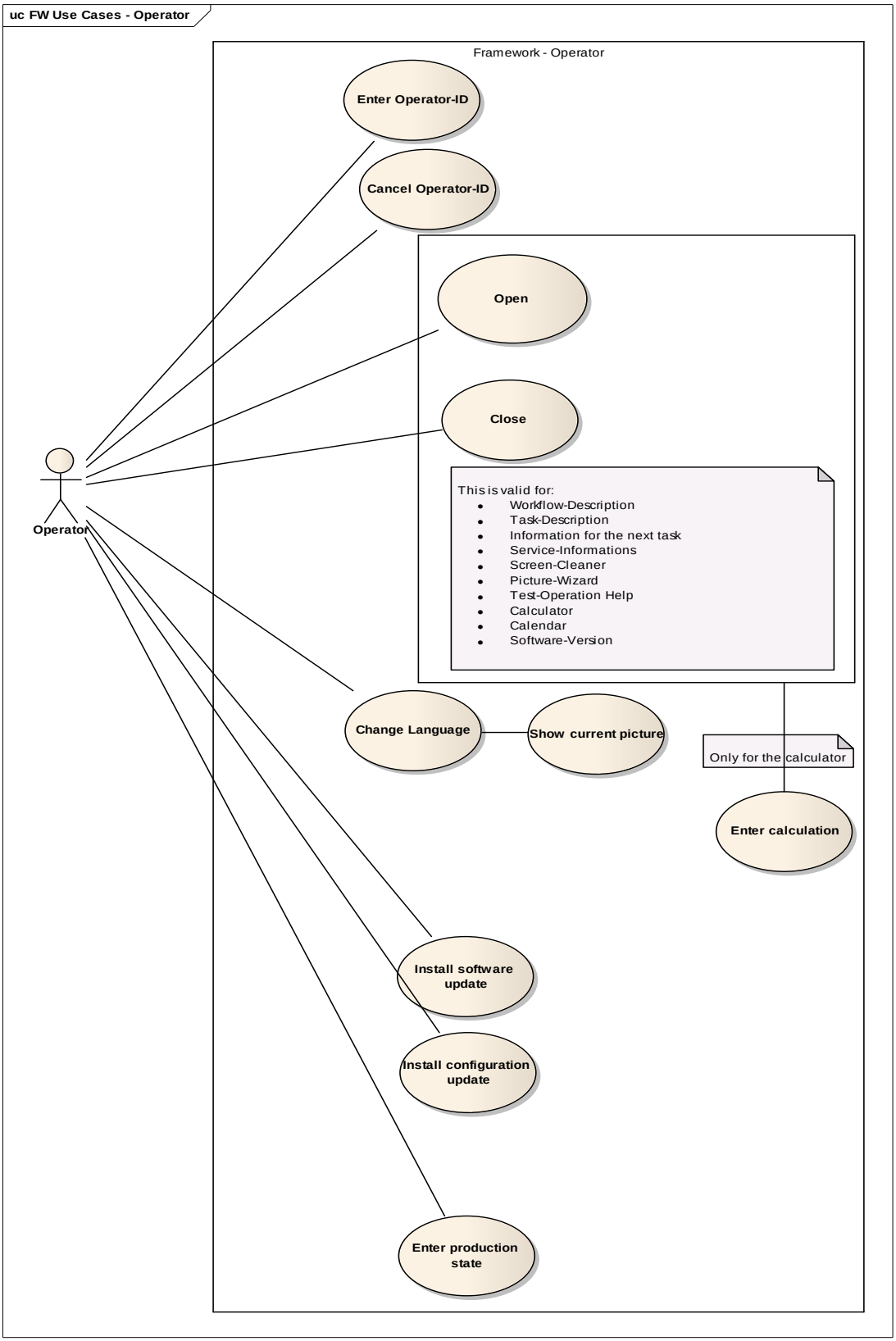


Abbildung 9: Anwendungsfallmodell der Operatorfunktionalität im Framework

4.4.2 Beschreibungen der Anwendungsfälle Entwickler

Aus den Tabellen im Anhang unter Kapitel 12.2.2.1 sind die Beschreibungen der Anwendungsfälle ersichtlich, die dem Akteur „Developer“ zugeordnet sind.

4.4.3 Beschreibungen der Anwendungsfälle System

Aus den Tabellen im Anhang unter Kapitel 12.2.2.2 sind die Beschreibungen der Anwendungsfälle ersichtlich, die dem Akteur „System“ zugeordnet sind.

4.4.4 Beschreibungen der Anwendungsfälle Operator

Aus den Tabellen im Anhang unter Kapitel 12.2.2.3 sind die Beschreibungen der Anwendungsfälle ersichtlich, die dem Akteur „Operator“ zugeordnet sind.

4.4.5 Synthese

Die folgende Tabelle 8 zeigt das Ergebnis der Zusammentragung von allen Informationen aus Lastenheft, Pflichtenheft, sonstigen Informationsquellen zum Framework, geordnet in die verschiedenen Elemente, um daraus das Fachklassendiagramm zu erstellen.

Tabelle 8: Synthese Framework

1. Klassen/Objekte	2. Attribute	3. Operationen
Adapter	Konfigurationsdatei (HW-Adapter) Konfigurationsdatei-Pfad	HW-Adapter laden
Menü	Menüpunkte	Menüpunkt hinzufügen
Fenster	Menü Menüpunkt	Fenster öffnen Fenster schliessen Fenster aktivieren Fenster deaktivieren
Dialog		Dialog öffnen Dialog schliessen Dialog aktivieren Dialog deaktivieren
Identifikationsnummer	Nummer	Überprüfen auf Operatornummer Überprüfen auf Auftragsnummer Überprüfen auf Bricknummer Überprüfen auf Ingotnummer Überprüfen auf Werkstückaufnehmernummer Identifikationsnummer aktivieren Identifikationsnummer deaktivieren Fehlermeldung anzeigen
Auftrag	Operator-Identifikationsnummer Auftrag-Identifikationsnummer Brick-Identifikationsnummer Ingot-Identifikationsnummer	Senden

	Werkstückaufnehmer-Identifikationsnummer Daten	
Antwort	Daten Url Auftragsinformationen Url Informationen zu nächsten Auftrag Url Wartungsinformationen	
Dokument	Url Typ (XML/HTML/Text)	Herunterladen Anzeigen
Sprache	Aktuelle Sprache	Sprache ändern Text anzeigen
Anmelden Dialog	Id Timeout	Dialog öffnen Abbrechen Bestätigen Id setzen Id senden Timer starten Eingabefeld fokussieren
Arbeitsbeschreibungshilfe		Fenster öffnen Anzeigen Fenster schliessen
Auftragsinformationen	Url	Fenster öffnen Anzeigen Fenster schliessen Herunterladen
Infos zum nächsten Auftrag	Url	Fenster öffnen Anzeigen Fenster schliessen Herunterladen
Wartungsinformationen	Url	Fenster öffnen Anzeigen Fenster schliessen Herunterladen
Hilfsmittel Dialog		Hilfsmittel aktivieren Hilfsmittel deaktivieren
Bildschirmreinigerhilfe	Timeout	Fenster öffnen Anzeigen Fenster schliessen Timer starten

Bilder-Assistent Dialog	Pfad	Dialog öffnen Anzeigen Dialog schliessen Nächstes Bild anzeigen Vorheriges Bild anzeigen
Testbedingungshilfe		Fenster öffnen Anzeigen Fenster schliessen Nächstes Bild anzeigen Vorheriges Bild anzeigen
Rechner Dialog	Operand 1 Operand 2 Resultat	Dialog öffnen Anzeigen Dialog schliessen Berechnen
Kalender Dialog		Dialog öffnen Anzeigen Dialog schliessen
SEMI E10 Produktionsstand	Status Timeout	Senden Ändern Timer starten
Software-Version		Dialog öffnen Anzeigen Dialog schliessen
Software-Update		Updates abfragen Update herunterladen Update installieren
Meldung Dialog		Dialog öffnen Anzeigen Dialog schliessen Bestätigen
Fehlermeldung Dialog		Dialog öffnen Anzeigen Dialog schliessen Bestätigen
Tastatur Dialog		Dialog öffnen Anzeigen Dialog schliessen Bestätigen
TouchScreen	Status	Symbol anzeigen

Barcode-Leser		Barcode-Leser-Daten einlesen Barcode-Leser-Daten speichern Barcode-Leser-Daten identifizieren Barcode-Leser-Daten aufsplitten
IS	Status Sendedaten Antwortdaten	Ist Bereit Daten senden Antwort abfragen

4. Multiplizitäten	5. Bedingungen/Periodizitäten	6. Assoziationen
beliebig viele Menüpunkte		

4.4.6 Fachklassendiagramm

Aus der Synthese ist das Fachklassendiagramm wie in der Abbildung 10 zu sehen ist, entstanden. Das Fachklassendiagramm zeigt die Klassen, die benötigt werden, um die aus dem Anwendungsfallmodel entstandenen Geschäftsanwendungsfälle abdecken zu können. Es entspricht deshalb nicht dem definitiven Klassendiagramm und wird im Detaildesign weiterentwickelt.

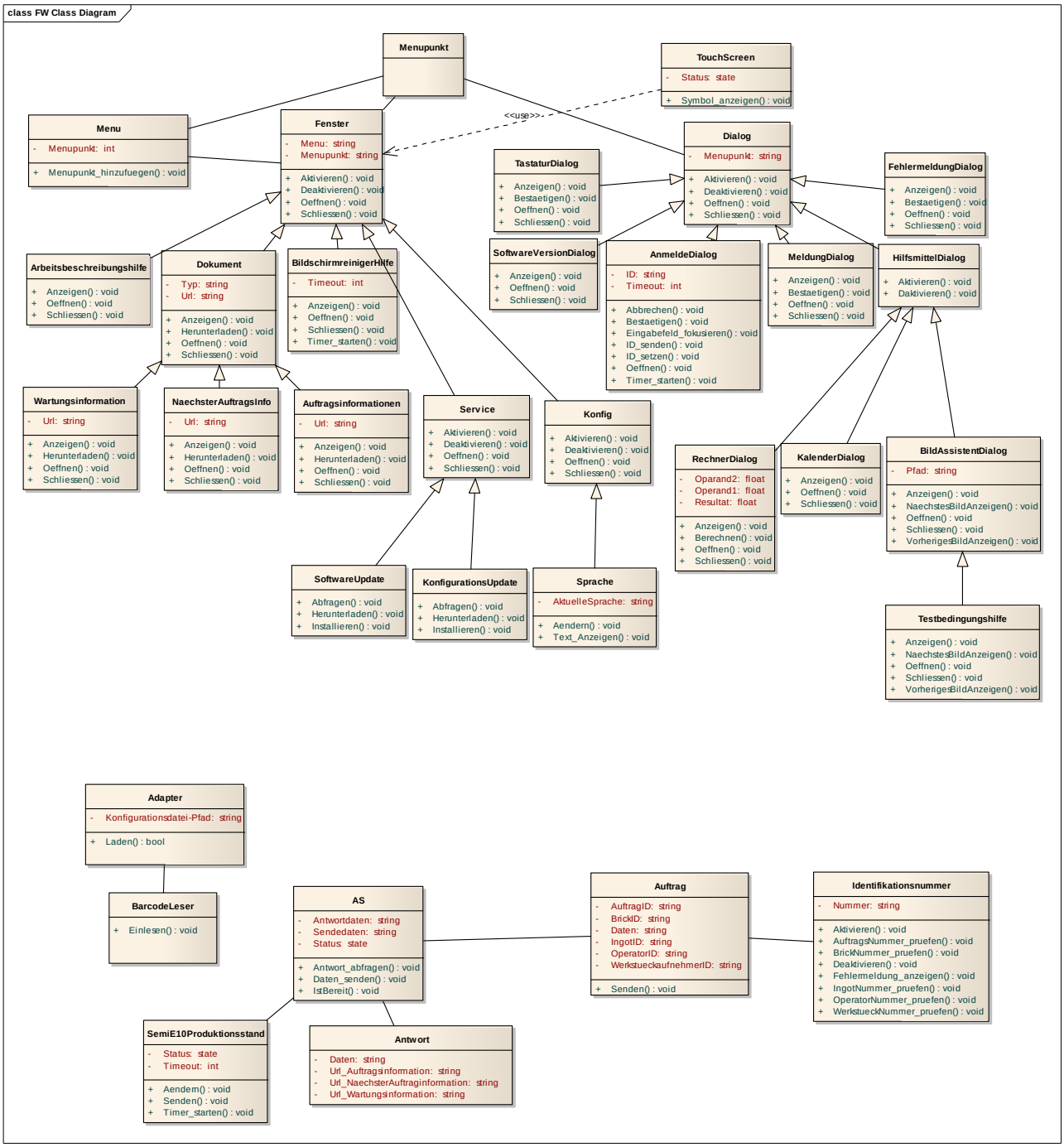


Abbildung 10: Fachklassendiagramm Framework

4.4.7 Beschreibung Fachklassendiagramm

Mit Hilfe der Synthese ist das Fachklassendiagramm in Abbildung 10 entstanden.

Darin enthalten sind die folgenden Klassen:

- Fenster/Menu/Menupunkt: Diese drei Klassen sind in dieser Form nicht mehr vorhanden im Detaildesign. Der Hauptgrund dafür ist die Problematik mit der WPF Technologie von Microsoft, wie in Kapitel 9.2.1 beschrieben ist. Es musste ein anderer Ansatz gefunden werden.

- Alle Klassen, die von der Klasse Fenster abgeleitet sind, sollen die Inhalte der einzelnen Screens und Dialoge darstellen.

- Die Klasse Adapter ist zuständig für das Laden der Adapter.

- Die Klasse AS (neu IS) zusammen mit Auftrag und Antwort bilden den Teil für die Kommunikation mit dem Integration Service.

- Die Klasse Identifikation ist zuständig für die Gültigkeitsüberprüfung der Identifikationsnummern.

4.5 Referenz-Anwendung „Brick Separierer“

4.5.1 Anwendungsfallmodell

Die Abbildung 11 zeigt das Anwendungsfallmodell des Brick Separierers.

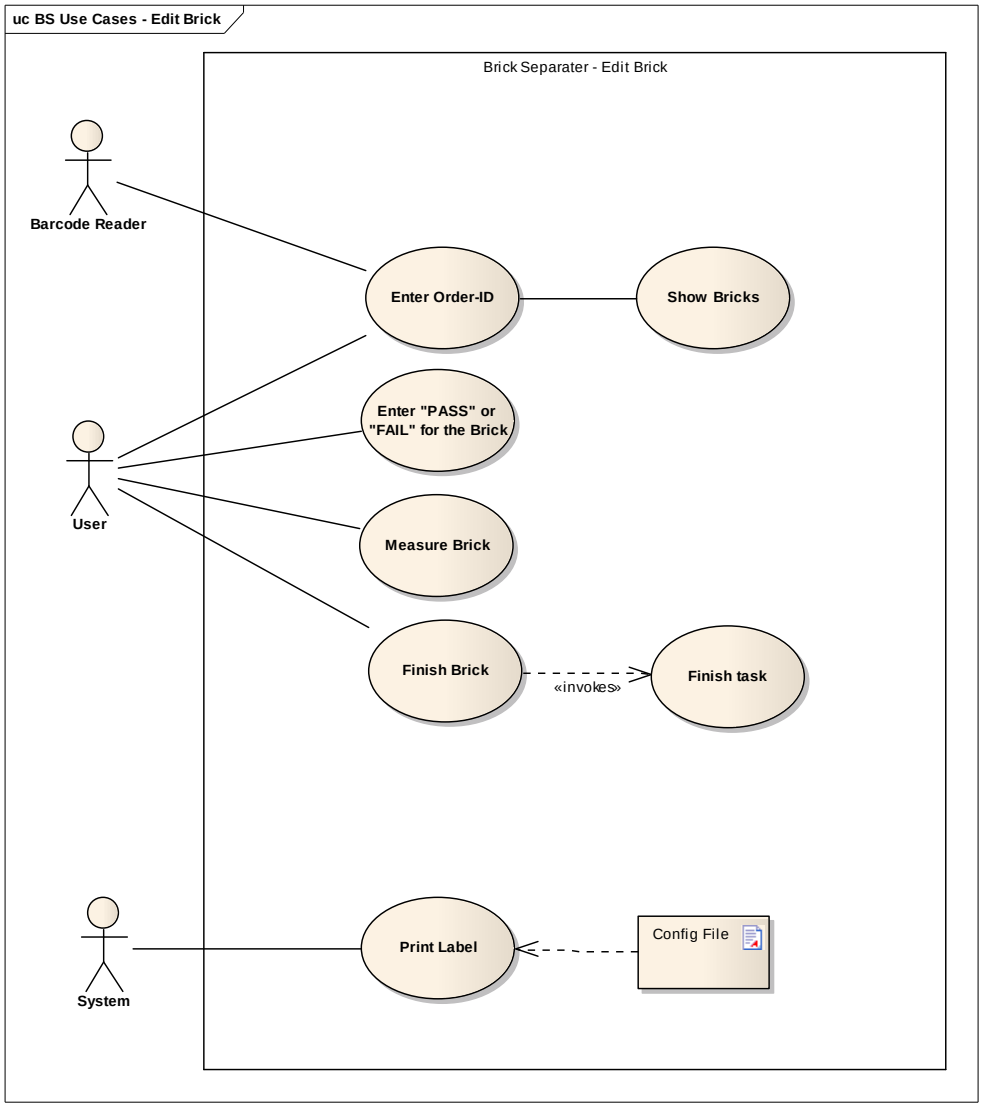


Abbildung 11: Anwendungsfallmodell für den Brick Separierer

Es ist zu sehen, dass die Auftrags-ID entweder per Barcode-Leser oder durch den Bediener getätigt werden kann. Anschliessend werden die Bricks angezeigt. Der Bediener kann einen Brick als „Pass“ oder „Fail“ markieren. Weiterere Anwendungsfälle sind das Messen, das Abschliessen eines Bricks und das Abschliessen eines Auftrages. Das System druckt anhand eines Templates die Etikette für den Brick aus.

4.5.2 Beschreibungen der Anwendungsfälle

Aus den Tabellen im Anhang unter Kapitel 12.2.3 sind die Beschreibungen der Anwendungsfälle ersichtlich.

4.5.3 Ablauf

In der folgenden Abbildung 12 ist der Gesamtablauf (Workflow) des Brick Separierers aufgezeigt.

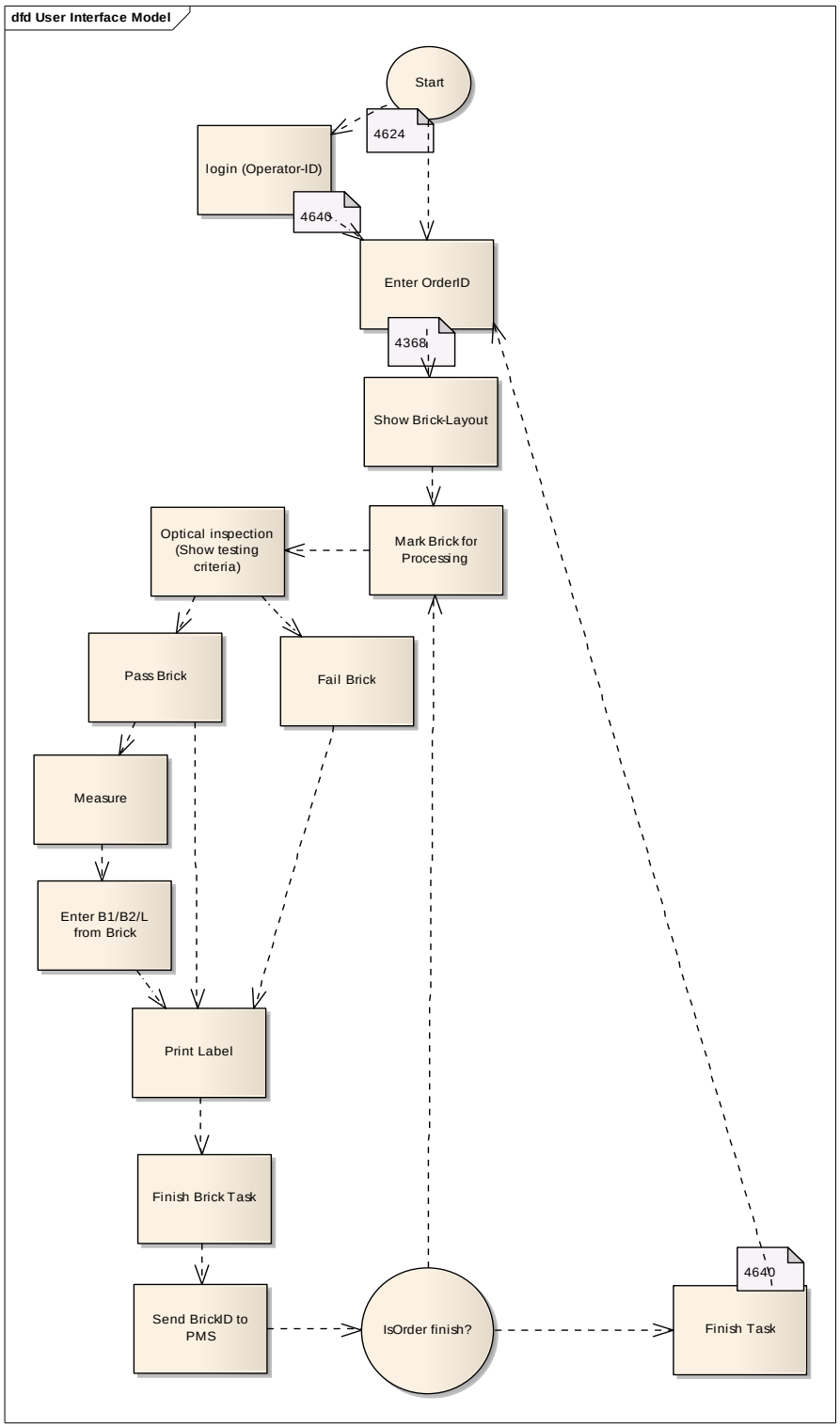


Abbildung 12: Ablauf Brick Separierer

Nach dem Starten der Anwendung wird zuerst der SemiE10 Status auf 4624 gesetzt. Optional kann sich der Bediener anhand seiner ID anmelden. Falls dies getan wird, wird der SemiE10 Status auf 4640 gesetzt. Nachdem die Auftrags-ID eingelesen und der SemiE10 Status auf 4368 gesetzt wird, werden die Bricks angezeigt und der zu bearbeitenden Brick wird markiert. Nun wird der Brick optisch inspiziert. Falls dieser in Ordnung („Pass“) ist, kann je nach Rezept eine Messung erfolgen oder es wird direkt eine Etikette gedruckt. Wird eine Messung durchgeführt, muss vor dem Ausdrucken der Etiketten die Länge, Breite und Höhe gemessen und ins Terminal eingetragen werden. Wird der Brick als „Fail“ markiert, wird direkt eine Etikette ausgedruckt. Anschliessend wird der Brick abgeschlossen, indem dies dem PMS mitgeteilt wird. Sobald der Brick fertig bearbeitet wurde, wird der nächste Brick verarbeitet. Ist kein weiterer Brick mehr zu bearbeiten, wird der Auftrag abgeschlossen und der SemiE10 Status auf 4640 wird gesetzt. Nun kann die nächsten Auftragsnummer eingegeben werden.

Da die Implementierung des Brick Separierers primär die Entwicklung eines GUI’s war, wurde auf ein Detaildesign verzichtet.

5 Prototyping

5.1 Hardware-Adapter

Für die Hardware-Adapter wurde je ein Prototyp für den Label-Drucker und den Barcode-Leser erstellt, um vor allem die Kommunikation mit den Geräten zu prüfen. Ebenfalls wichtige Prüfkriterien waren die Konfiguration mit .NET Mechanismen, das dynamische Laden der erstellten Adapter-Bibliotheken, sowie das Integrieren in eine Testanwendung.

5.2 Framework

Für das Framework wurde nicht direkt ein Prototyp erstellt. Vielmehr wurde vor allem mit der neuen WPF Technologie experimentiert, um herauszufinden, was damit möglich ist. Aus diesen Erkenntnissen haben sich bereits die ersten Probleme herauskristallisiert, die auch im Kapitel 9.2.1 beschrieben sind.

5.3 Referenz-Anwendung „Brick Separierer“

Da für das Erstellen der Terminal-Anwendung „Brick Separierer“ bereits auf das Framework zurückgegriffen wird und keine weitere komplexe und unsichere Funktionalität enthält, wurde entschieden, kein Prototyp zu erstellen.

6 Design

6.1 Hardware-Adapter

Die Grundfunktionalität mit den Basisklassen und Interfaces ist ins Framework integriert.

6.1.1 Label-Drucker

Im Anhang unter Kapitel 12.4.1.1 ist das Klassendiagramm für den Label-Drucker abgebildet.

Es unterscheidet sich in der Architektur und Funktionalität nur minimal zum Fachklassendiagramm aus dem Grobdesign/Analyse im Kapitel 4.3.2.4.

6.1.2 Barcode-Leser

Im Anhang unter Kapitel 12.4.1.2 ist das Klassendiagramm für den Barcode-Leser abgebildet.

Es unterscheidet sich in der Architektur und Funktionalität nur minimal zum Fachklassendiagramm aus dem Grobdesign/Analyse im Kapitel 4.3.2.4.

6.2 Framework

6.2.1 Paketaufteilung

Das Framework ist aufgeteilt in fünf Packages, bzw. Namespaces, die in unten beschrieben werden.

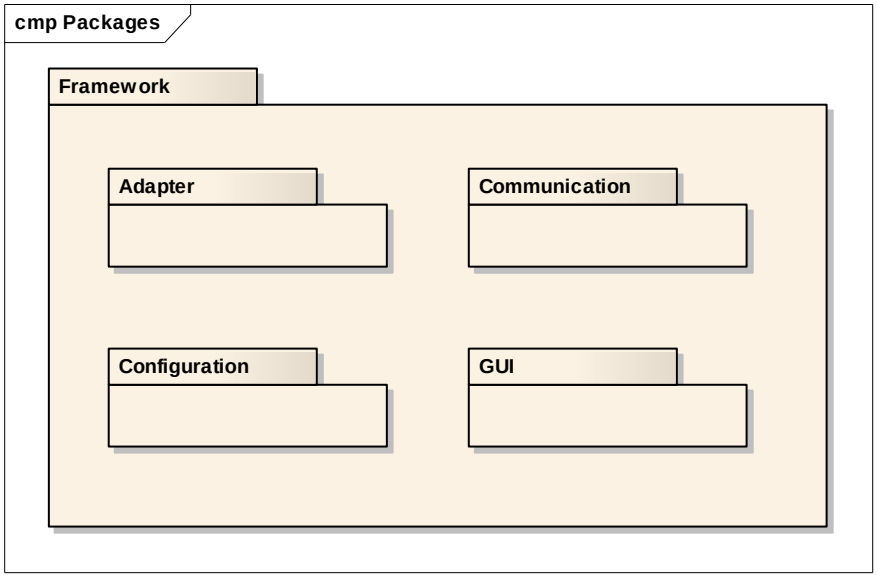


Abbildung 13: Paketaufteilung Framework

6.2.1.1 Hauptpaket Framework

Im Hauptpaket sind alle globalen Aufgaben enthalten. Dazu gehören folgende Komponenten:

- die Hauptklasse
- die Verwaltung des Operators
- die Speicherung der aktuellen Auftragsdaten
- die Sprachumschaltung
- Software Update Handling
- SemiE10 Status
- Allgemeine Hilfsklassen

6.2.1.2 Paket Adapter

Das Paket *Adapter* enthält die Interfaces, Enums und Klassen für das Ansprechen der Hardware-Adapter.

Die Hardware-Adapter selbst bedienen sich ebenfalls bei diesem Paket. Einerseits bei der Entwicklung eines Adapters, andererseits zur Laufzeit.

6.2.1.3 Paket Communication

Im Paket *Communication* sind die Clients untergebracht, um mit dem PMS zu kommunizieren. Das sind:

- ISClient: Der ISClient kommuniziert lokal mit dem Integration-Service, der als Schnittstelle zum PMS dient.
- PMSClient: Dieser Client funktioniert als Web Browser und dient dem Download von Dokumenten und Rezepten ab dem PMS Web Server.

6.2.1.4 Paket Configuration

In diesem Paket ist die gesamte Konfigurationsverwaltung enthalten.

Sämtliche Konfigurationselemente sind in so genannten Sektionen untergebracht. Alle Sektionen in der Konfiguration sind in Sektionsgruppen eingeteilt. Für jede Sektionsgruppe existiert eine eigene Klasse. Dazu existiert auch für jede Sektion eine eigene Klasse.

Der Zugriff auf die Konfigurationselemente erfolgt über den ConfigManager.

Für die Speicherung von Benutzereinstellungen in der Registrierungsdatenbank dient der RegistryManager, der ebenfalls in diesem Paket untergebracht ist.

6.2.1.5 Paket GUI

Das grösste Paket im Framework ist das Paket GUI. Es ist zusätzlich unterteilt in Untergruppen. Direkt im Paket eingeteilt sind folgende Teile:

- Die Basisklassen für die Fenster, die virtuellen Tastatur Klassen und die Inhaltsklassen
- Alle im Framework integrierten Inhaltsklassen wie z.B. Config, MessageDialog, Help
- Die Styles für die darin enthaltenen Inhaltsklassen

Folgende Untergruppen sind vorhanden:

- Button: Hier in dieser Untergruppe sind die zwei Arten von Buttons implementiert sowie die dazugehörende Basisklasse. Zusätzlich sind auch Hilfsklassen für die Buttons und die Styles enthalten.
- Dialog: Die Untergruppe Dialog enthält den Grundaufbau der Dialoge
- InputField: Diese Untergruppe enthält die Basisklasse für ein Eingabefeld und zwei abgeleitete Klassen für die Eingabe eines numerischen Wertes und eines Textes.
- Screen: Die Untergruppe Screen enthält den Grundaufbau der Screens
- Utilities: Hier sind die geforderten Hilfsmittel implementiert. Das sind: Rechner, Kalender, Bild-Assistent, Testbedingungshilfe und Bildschirmreiniger. Zusätzlich beinhaltet es die Inhaltsklasse für die Utilities.

6.2.2 GUI Architektur

Dem Aufbau des GUI liegt die WPF (Windows Presentation Foundation) zugrunde.

6.2.2.1 Fenster

Das Framework bietet zwei Arten von Fenstern:

Die Klasse Screen stellt das Vollbild Fenster zur Verfügung, in dem der Grundaufbau enthalten ist. In der Mitte ist der Platz für den Inhalt. Die Belegung der Schaltflächen auf den Seiten und am unteren Rand können beim entsprechenden Screen Eintrag in der Konfigurationsdatei definiert werden. Drei Arten sind möglich: ein weiterer Screen, ein Dialog oder eine Funktion.

Die Klasse Dialog stellt das Dialog Fenster mit Rahmen und einer „ButtonBox“ am unteren Rand zur Verfügung. In der Mitte ist der Platz für den Inhalt. Die Art der „ButtonBox“ kann beim entsprechenden Dialog Eintrag in der Konfigurationsdatei definiert werden. Die Auswahl an Typen ist in Enum EDialogButtons definiert.

6.2.2.2 Inhaltsklassen

Jeder in der Konfigurationsdatei definierter Screen oder Dialog wird mit einem Namen versehen. Dieser Name ist auch gleich der Name der Inhaltsklasse. Beim Erzeugen eines Screens oder Dialogs wird mit diesem Namen über die Reflection eine Instanz der Inhaltsklasse erzeugt und in das Fenster eingefügt.

Dadurch ist es möglich eine Inhaltsklasse einem Screen, sowie einem Dialog zuzuordnen. Dies erfolgt mit der entsprechenden Definition in der Konfigurationsdatei.

6.3 Referenz-Anwendung “Brick Separierer”

Die Referenzanwendung besteht aus fünf Klassen, welche wie folgt beschrieben werden:

- Klasse „Program“: Darin befindet sich die Main-Methode, die zum Starten der Anwendung dient.
- Klasse „Top“: Zum Einen befindet sich darin die Logik für das GUI des Top-Bildes definiert und zum Anderen die Logik des Prozessablaufs des Brick Separierers.
- Klasse „Measure“: Darin ist die Logik für das GUI des Measure-Bildes definiert.
- Klasse „OrderID“: Darin ist die Logik für das GUI des OrderID-Bildes definiert.
- Klasse „Brick“: Diese Klasse beinhaltet die Daten der Bricks, die aus dem Rezept ausgelesen werden, plus den CustomButton, der für die Anzeige im GUI gebraucht wird.

Im Kapitel 12.4.3 ist das Klassendiagramm ersichtlich. Darin sind die fünf Klassen abgebildet. Die wichtigste Klasse ist die Top-Klasse, darin befindet sich die ChangeStep(ESteps)-Methode, worin die Abläufe des Bricks Separierers implementiert sind.

6.4 API

Alle Klassen, Interfaces und Enums sind direkt im Quellcode beschrieben und in einer generierten API Dokumentation zusammengefasst. Für folgende Komponenten sind API Dokumentationen vorhanden:

- Label-Drucker
- Barcode-Leser
- Framework

Die API Dokumentation sind im HTML Format vorhanden und befinden sich im Unterverzeichnis doc des entsprechenden Visual Studio Projektes.

7 Test

Mit der gewählten Entwicklungstechnik XP (eXtreme Programming) wurden während der gesamten Entwicklungszeit in der Art “Vorbeugung” bereits viele “Fehler” behoben oder erst gar nicht gemacht.

Trotzdem konnte nicht ganz auf Tests verzichtet werden.

7.1 Hardware-Adapter

7.1.1 Unit-Test

Die Adapter beinhalten nur minimale Logik.

Mit dem Einverständnis des Auftraggebers wurde deshalb vorerst auf Unit-Tests verzichtet.

7.1.2 Test Szenarien

Im Anhang unter dem Kapitel 12.3.1 sind alle Test Szenarien aufgelistet, die zusätzlich als Kontrolle der Anwendungsfälle geprüft wurden.

7.1.3 Test-Anwendungen

Um die Entwicklung der Adapter auf Lauffähigkeit und Funktionalität hin laufend testen zu können, wurde für die beiden Adapter jeweils eine kleine Test-Anwendung erstellt.

Es wurde bei diesen zwei Test-Anwendungen darauf geachtet, dass die Adapter genau so bedient werden, wie sie auch innerhalb des Frameworks zum Einsatz kommen. Das heisst, der entsprechende Adapter wird dynamisch geladen und mittels der vorgesehenen Methode gestartet.

7.1.4 Abnahme

Als Abschlusstest für die beiden Hardware-Adapter dienten das Dokument „Abnahme Der Hardwareadapter.doc“, der Design und der Quellcode. Die Abnahme erfolgte durch Arjan DJ Paijralbe.

Getestet wurden folgende Punkte zur Grundstruktur der Hardware-Adapter:

- Wurde C# verwendet
- Programmierrichtlinien eingehalten
- Ist das Umschreiben eines Adapters auf ein anderes Modul innerhalb von 1-1.5 Tagen möglich
- Hat es Beschreibung und Beispiele zu den Adaptern
- Hat der Adapter Init und Exit Methoden
- Wie sieht der Fehlermechanismus aus
- Kann der Adapter konfiguriert werden

Getestet wurden folgende Punkte zum Adapter Barcode-Leser:

- Wird die Grundstruktur verwendet

- Sind die Programmierrichtlinien eingehalten worden
- Wird eine Verbindung zur Schnittstelle automatisch aufgebaut
- Erscheint ein Text beim Einlesen eines Barcodes
- Ist eine andere COM Schnittstelle benutzbar

Getestet wurden folgende Punkte zum Adapter Label-Drucker:

- Wird die Grundstruktur verwendet
- Sind die Programmierrichtlinien eingehalten worden
- Wird eine Verbindung zur Schnittstelle automatisch aufgebaut
- Werden die Daten ab Vorlage korrekt ausgegeben
- Ist eine andere COM Schnittstelle benutzbar
- Test mit drei verschiedenen Vorlagen (Ascii/UTF-8/UFT-16)

7.2 Framework

7.2.1 Unit-Test

Das Framework beinhaltet nur minimale Business-Logik und stellt vor allem GUI Komponenten zur Verfügung.

Test eines GUI ist in der Regel äusserst aufwendig und nicht unbedingt „messbar“.

Mit dem Einverständnis des Auftraggebers wurde deshalb auf Unit-Tests verzichtet.

7.2.2 Test Szenarien

Im Anhang unter dem Kapitel 12.3.2 sind alle Test Szenarien aufgelistet, die zusätzlich als Kontrolle der Anwendungsfälle geprüft wurden.

7.2.3 Test-Anwendung

Um die Entwicklung des Frameworks auf Lauffähigkeit und Funktionalität hin laufend testen zu können, wurde dafür eine Test-Anwendung erstellt.

Es wurde darauf geachtet, dass diese Anwendung möglichst alle Anforderungen aus dem Pflichtenheft abdeckt.

7.2.4 Abnahme

Die Abnahme erfolgte als Walkthrough und erfolgte durch Arjan DJ Paijralbe.

Als Abnahmedokument diente das Pflichtenheft, das Design, der Quellcode und die API Dokumentation.

Getestet wurden sämtliche, im Pflichtenheft aufgeführten spezifizierten Anforderungen. Getestet wurden auch Punkte, wie sie bei der Abnahme der Hardware-Adapter durchgeführt wurden.

7.3 Referenz-Anwendung “Brick Separierer”

7.3.1 Unit-Test

Die Referenz-Anwendung beinhaltet nur minimale Business-Logik und stellt vor allem GUI Komponenten zur Verfügung.

Mit dem Einverständnis des Auftraggebers wurde deshalb auf Unit-Tests verzichtet.

7.3.2 Test

Test eines GUI ist in der Regel äusserst aufwendig und nicht unbedingt „messbar“. Die Anwendung wurde einem Anwender mit Kenntnissen vom Prozess und einem, welcher den Prozess nicht kennt, zum Testen gegeben. Die daraus resultierenden Erkenntnisse wurden im GUI korrigiert.

7.3.3 Abnahme

Die Abnahme erfolgte als Walkthrough und durch Arjan DJ Paijralbe erfolgte.

Als Abnahmedokument diente das Pflichtenheft, das Design, der Quellcode und die API Dokumentation.

Getestet wurden die im Pflichtenheft aufgeführten spezifizierten Anforderungen.

7.4 Integrationstest

Der Integrationstest bestand darin, die erstellte Software auf der dafür vorgesehenen Hardware auszuführen.

Für die Kommunikation mit dem IS (Integration Service) musste zu Testzwecken zusätzlich ein einfacher IS-Simulator implementiert werden, da der dafür vorgesehene IS bis zum Ende der Diplomarbeit nicht zur Verfügung stand.

Für die Kommunikation mit dem PMS Web-Server musste zu Testzwecken auf einen anderen Web-Server zurückgegriffen werden, da der dafür vorgesehene Web-Server aus der PMS Testumgebung bis zum Ende der Diplomarbeit nicht zur Verfügung stand.

Auf den Web-Server wurden folgende Dokumente hochgeladen:

- Druck-Vorlage für den Hardware-Adapter Label-Drucker
- HTML Dokumente für: Auftragsinformationen, Informationen zu nächsten Aufträgen, Wartungsinformationen
- Rezept für den Brick Separierer

7.4.1 Hardware

Folgende Hardware stand für den Integrationstest zur Verfügung:

- Siemens Panel PC 477 B, ohne Touchscreen, mit Folientastatur, mit 1 COM Schnittstelle
- Winmate T-series Panel PC, mit Touchscreen, mit 3 COM Schnittstellen
- Barcode-Leser Powerscan D8330 von Datalogic

Der Test mit einem Label-Drucker konnte nicht durchgeführt werden, da ein solches Gerät nicht zur Verfügung stand. Die Ausgabe des Label-Drucker Adapters wurde deshalb zu Testzwecken auf einem der Entwicklungs-Laptops in der Hyper-Terminal Anwendung angezeigt. Das Terminal wurde dazu mittels eines RS232 Kabel mit dem Laptop verbunden.

Zusätzlich wurden folgende Komponenten eingesetzt:

- USB zu RS232 Konverter (der Siemens Panel PC hat nur 1 COM Schnittstelle)
- Normale Tastatur (für den Winmate Panel PC)

7.4.2 Software

Auf den beiden verfügbaren Panel PC's ist das Betriebssystem Windows XP Embedded installiert.

Um die Terminal-Anwendung ausführen zu können, musste vorgängig das Microsoft .NET Framework Version 3.5 installiert werden.

7.4.3 Vorgehen

Der Integrationstest wurde mit den folgenden Anwendungen durchgeführt:

- Testanwendung TestFrameworkApp
- Referenz-Anwendung „Brick Separierer“

Dabei wurden sämtliche in den Anwendungen integrierten Bestandteile aufgerufen und auf deren Funktionalität getestet, inklusive der Kommunikation mit der angeschlossenen Hardware, dem IS (Integration Service) und dem Web-Server.

8 Resultate

8.1 Hardware-Adapter

8.1.1 Abnahme

Die Abnahme der beiden Hardware-Adapter ist gemäss Kapitel 7.1.4 durchgeführt worden und hat zu folgendem Ergebnis geführt:

Die Abnahmepunkte haben verschiedene Wertungen. Dabei ist darauf geachtet worden, dass die Punkte mit der Wertung „Zwingend“ unbedingt erfolgreich sein müssen.

Die in Kapitel 7.1.4 aufgeführten Punkte sind getestet und der Abnahmetest ist insgesamt als erfolgreich bewertet worden.

8.1.2 Label-Drucker

Der in der Diplomarbeit vorgesehene Drucker konnte nicht rechtzeitig zur Verfügung gestellt werden, deshalb wurde beschlossen, dass der Drucker als solches von der Diplomarbeit gestrichen wird. Die Funktion des Hardware-Adapters wurde getestet, indem die Daten an einem anderen Computer am Hyperterminal ausgegeben wurden.

8.2 Framework

Anhand der folgenden Abbildung wird gezeigt, wie die Umsetzung des GUI's vom Framework gestaltet wurde:

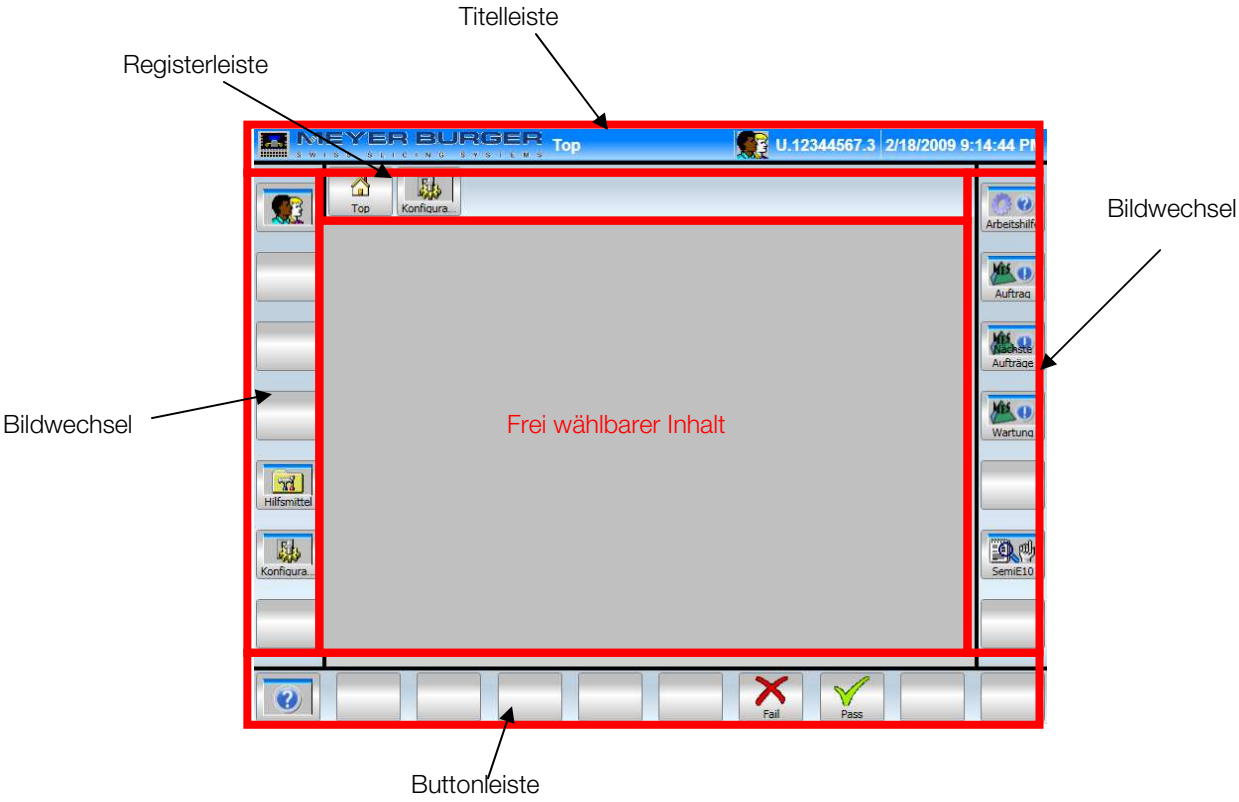


Abbildung 14: GUI Grundelemente Framework

Das Framework besteht aus den folgenden Teilen, welche den frei wählbaren Inhalt einer Anwendung umrahmen und mit der benötigen Funktionalitäten versieht:

- Die Titelleiste, welche das Firmenlogo, den Benutzername, die Benutzer-ID, sowie das Datum und die Uhrzeit beinhaltet.
- Die Registerleiste beinhaltet Funktionen, wie das Zurückkehren auf das Hauptbild.
- Die Leisten für den Bildwechsel sind dazu da, um in einer Applikation Bildwechsel zu gewährleisten. Standardmässig sind im Framework bereits folgende Bildwechsel integriert:
    - Hilfsmittel, wie ein Rechner, ein Kalender, ein Bilderassistent, eine Anzeige für Testbedingungen und einem Bildschirmreiniger.
    - Konfiguration, worin ein Software-Update oder eine Sprachumschaltung getätigt werden kann.
    - Arbeitshilfe, worin anhand eines Dokumentes, das vom PMS zur Verfügung gestellt wird, beschrieben wird, wie die Arbeit erledigt werden soll.

- Auftrag, worin anhand eines Dokumentes, das vom PMS zur Verfügung gestellt wird, Daten zum aktuellen Auftrag angezeigt werden können.
- Nächster Auftrag, worin anhand eines Dokumentes, das vom PMS zur Verfügung gestellt wird, Daten zum nächsten Auftrag angezeigt werden können.
- Wartung, wo anhand eines Dokumentes, das vom PMS zur Verfügung gestellt wird, Wartungsdaten angezeigt werden können.
- SemiE10, wo der SemiE10 Status übersteuert werden kann.

- In der Buttonleiste befinden sich die Buttons, die als Funktionstasten frei belegt werden können.

Das Framework bietet an, dass in der Konfigurations-Datei eingestellt werden kann, ob ein Touch-Screen oder eine Tastatur mit Maus vorhanden ist. Sobald ein Touch-Screen konfiguriert wird, erscheint neben Eingabefeldern ein Icon mit einer Tastatur darauf. Drückt man auf diese Taste erscheint ein virtuelles Numpad oder eine Tastatur, jenachdem, ob es sich um eine numerische oder alphanumerische Eingabe handelt. Die folgende Abbildung zeigt ein Bespiel für das virtuelle Numpad:

Abbildung 15: Virtuelle Tastatur (Numpad)

8.2.1 Software Update

Im Einverständnis mit dem Auftraggeber wurde auf die Trennung der Updates von Applikation und Konfiguration verzichtet. Ein Software Update beinhaltet nun beides, die Applikation mit all den benötigten Bibliotheken, Zusatztabellen und die Konfigurationsdatei.

Die gesamte Funktionalität für das Software Update der Anwendung wurde in eine externe Konsolenanwendung, den so genannten Updater ausgelagert.

8.2.2 Abnahme

Die Abnahme des Frameworks ist gemäss Kapitel 7.2.4 durchgeführt worden und hat zu folgendem Ergebnis geführt:

Nicht alle Punkte konnten direkt getestet werden. Diese fehlenden Punkte werden mit dem Abnahmetest der Referenz-Anwendung behandelt.

Die in Kapitel 7.2.4 aufgeführten Punkte sind getestet worden und der Abnahmetest ist insgesamt als erfolgreich bewertet.

8.3 Referenz-Anwendung “Brick Separierer”

Für die Referenzanwendung „Brick Separierer“ muss, wie in der folgenden Abbildung gezeigt, eine Auftrags-ID eingegeben werden. Ist diese korrekt, wird vom PMS ein Rezept heruntergeladen, worin die benötigten Informationen abgelegt sind.

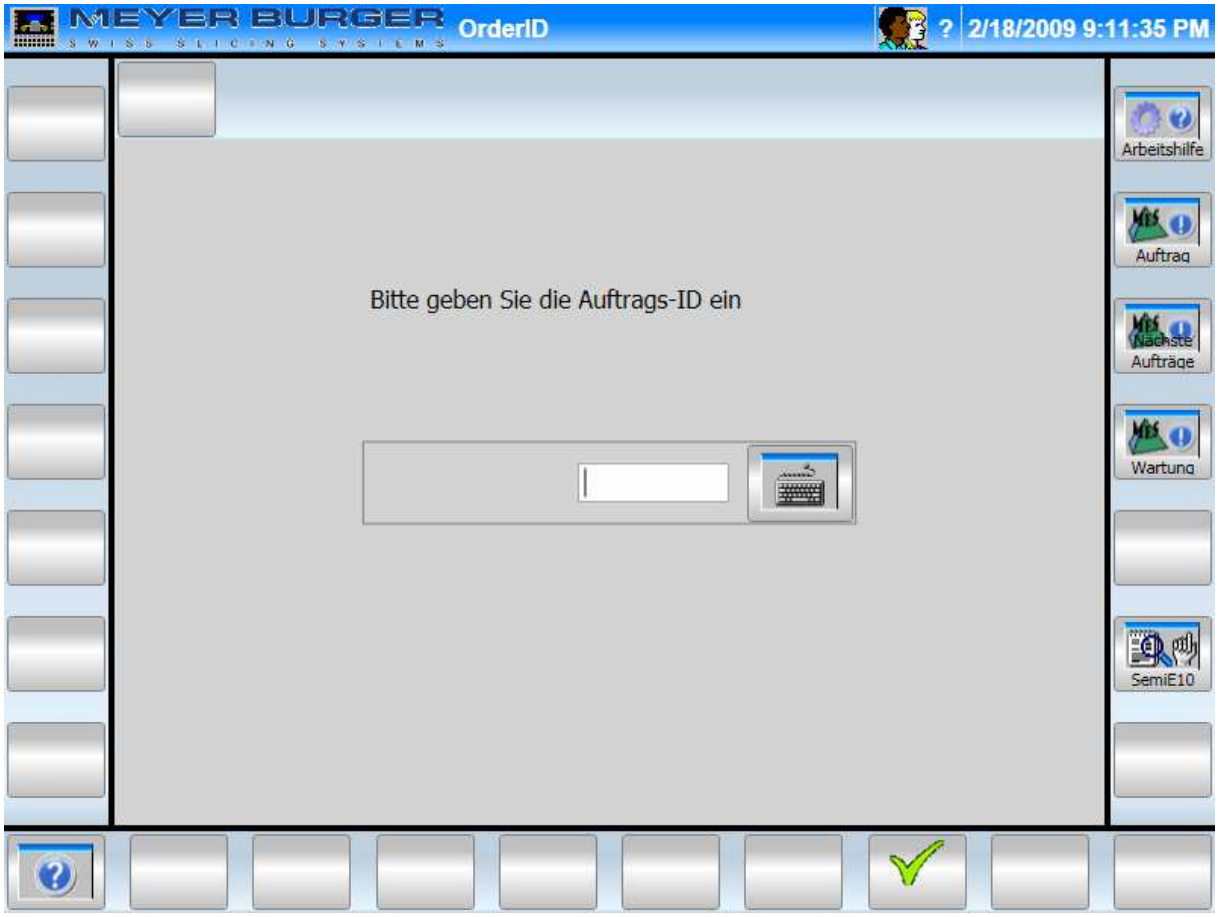


Abbildung 16: Bild Eingabe Auftrags-ID

Anhand dieses Rezeptes werden, wie in der nachfolgenden Abbildung gezeigt, die Bricks angezeigt und markiert. Die Beschriftung A1,A2,…,B1,B2,…. wird vom Rezept vorgegeben. Die echten Bricks kommen mit derselben Beschriftung vom vorherigen Arbeitsplatz zu dieser Station. Mittels der Buttons „Pass“ und „Fail“ können die Bricks als gut oder schlecht befunden werden. Die Farben der Bricks zeigen den Zustand des Bricks an und sind wie folgt definiert:

- Grau: Der Brick ist nicht bearbeitet
- Gelb: Markiert den aktuell zu bearbeitende Brick
- Grün: Pass
- Rot: Fail

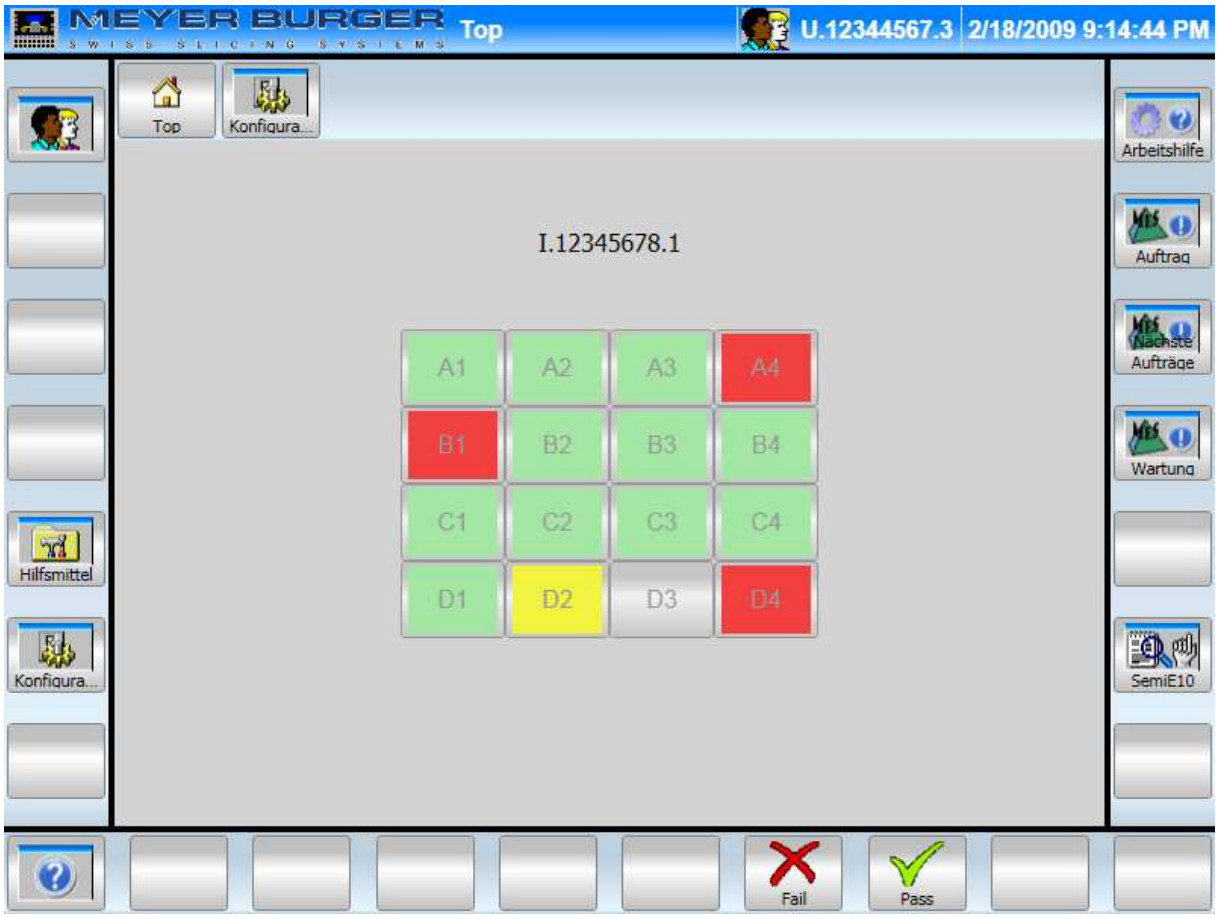


Abbildung 17: Bild Übersicht Bricks

Muss der Brick anhand des Rezeptes vermessen werden, wird automatisch das folgende Bild angezeigt:

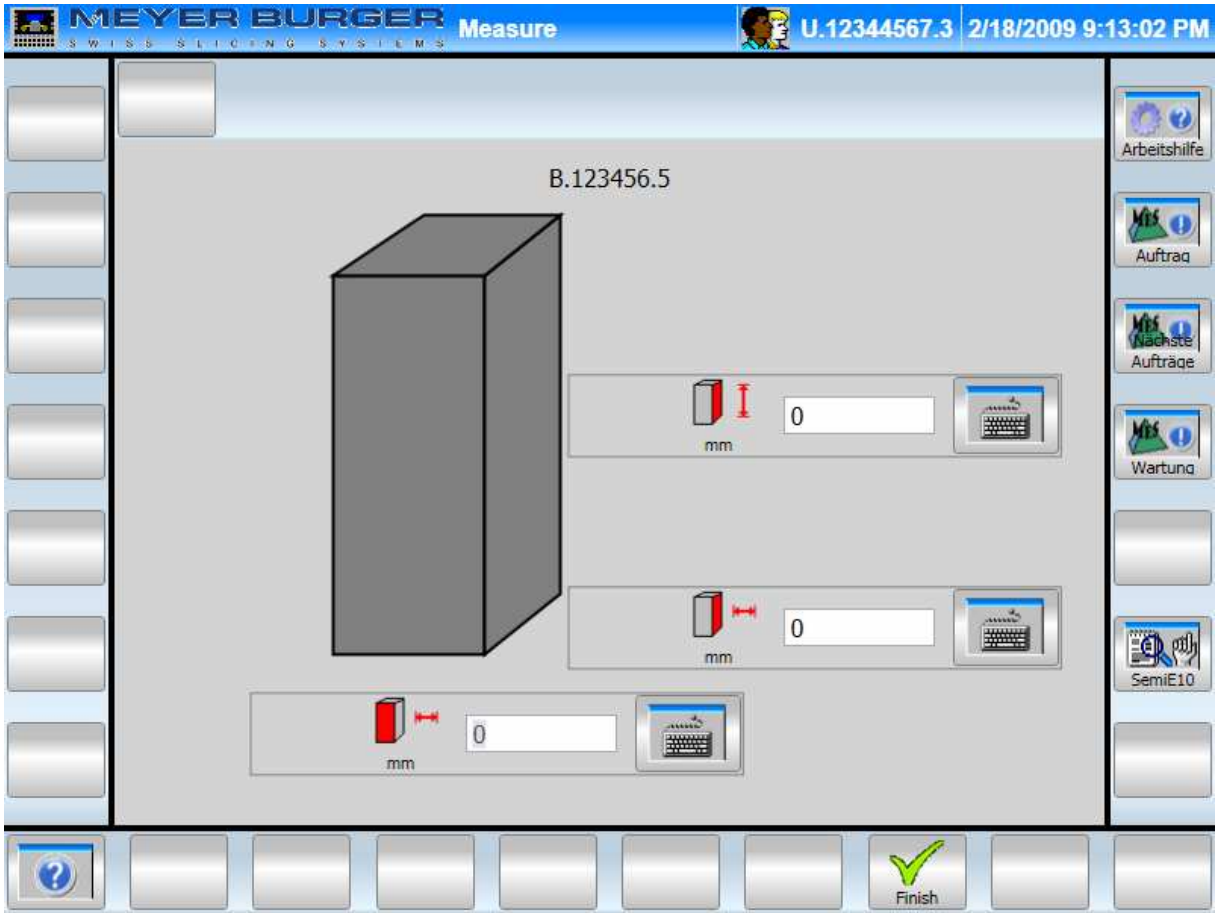


Abbildung 18: Bild Messung eines Bricks

In diesem Bild kann der Bediener die gemessenen Werte eintragen und mit dem „Finish“-Button den aktuellen Brick abschliessen.

8.3.1 Abnahme

Die Abnahme erfolgte als Walkthrough und erfolgte durch Arjan DJ Paijralbe.

Getestet wurden sämtliche, im Pflichtenheft aufgeführten spezifizierten Anforderungen. Getestet wurden auch Punkte, wie sie bei der Abnahme der Hardware-Adapter durchgeführt wurden.

Als Abnahmedokument diente das Pflichtenheft, das Design, der Quellcode und die API Dokumentation.

8.4 Dokumentation

8.4.1 “Entwicklerhandbuch”

Da das Framework bereits zu einer nicht zu unterschätzenden Grösse herangewachsen ist, ist es unerlässlich, eine Dokumentation für das Erstellen von weiteren Terminal-Anwendungen zu kreieren.

Mit dem Ende der Diplomarbeit ist noch lange nicht Schluss mit dem Gesamtprojekt PMS und den diversen Teilprojekten wie das TerminalPMS. Von daher macht es keinen Sinn, bereits jetzt eine solche Dokumentation zu erstellen.

8.4.2 API

Die API-Dokumentation ist für folgende Quellcode-Projekte vollumfänglich vorhanden:

- Hardware-Adapter Barcode-Leser
- Hardware-Adapter Label-Drucker
- Framework

Die Dokumentierung des Quellcodes ist direkt im Quellcode integriert und ist mit dem Standardmechanismus des Visual Studios formatiert.

Zur Generierung der API Dokumentation wurde die Software Doxygen verwendet.

8.5 Integration

8.5.1 Integrationstest

Die in Kapitel 7.4 aufgeführten Punkte wurden getestet und der Integrationstest als erfolgreich bewertet.

8.6 Projektplan

Der Projektplan (siehe Anhang, Kapitel 12.1) konnte mit leichter Verspätung eingehalten werden. Die Verspätungen resultierten daraus, dass zu Beginn und während der Arbeit notwendige Informationen nicht oder nur schwer eingeholt werden konnten. Hinzu kam, dass keine Anbindung an den IS oder ans PMS erstellt werden konnte, da diese bis anhin nicht funktionsfähig waren, was dazu führte, dass selber ein Dummy-IS erstellt wurde. Leider konnte der Label-Drucker nicht rechtzeitig beschafft werden, was die Abnahme verzögerte.

9 Diskussion

Insgesamt verlief das Projekt gut. Wir konnten alle Muss-Anforderungen erfüllen. Aufgrund der sehr guten Programmierkenntnisse von Herrn Beat Bösiger konnten grössere Schwierigkeiten vermieden werden.

9.1 Projektablauf

Im gesamten Ablauf des Projektes wurde darauf geachtet, möglichst genau nach Vorgabe aus dem Unterricht des MAS-IT Studiums vorzugehen.

Als absolutes Muss zu bewerten, ist das Pflichtenheft mit den genau spezifizierten Anforderungen des Produktes. Das Pflichtenheft war Gold wert und hat uns eine Menge Ärger erspart. Es hat uns als Absicherung bestens gedient.

Ebenfalls absolut sinnvoll ist ein zeitlicher Projektplan. Dadurch ist stets ersichtlich, wo man gerade steht.

Durch die oben genannten Punkte ist es uns gelungen, ohne grössere Schwierigkeiten den Termin halten zu können und die Arbeit fertig abliefern zu können.

Die erlernte und angewendete Vorgehensweise hat uns gezeigt, dass es durchaus sinnvoll ist, sich gut zu organisieren und möglichst nach einem bewährten Schema vorzugehen.

9.2 Framework

9.2.1 WPF

Mit der gewählten WPF-Technologie von Microsoft traten diverse Schwierigkeiten auf. Da dies eine relativ neue Technologie ist, traten mehrmals Schwierigkeiten auf, resp. Kinderkrankheiten kamen zum Vorschein. Einige Probleme konnten bereinigt werden, indem das Service Pack 1 für's Visual Studio 2008 installiert wurde. Andere Probleme mussten technisch umgangen werden.

Für ein nächstes Projekt würden wir nicht mehr WPF für das GUI einsetzen. Die Möglichkeiten schränken zu sehr ein. Vor allem, was objekt-orientierte Programmierung angeht.

Hier zwei Beispiele:

- Eine Klasse (XAML) mit WPF Technologie kann nicht von einer abstrakten Klasse abgeleitet werden. Es wurden zwar Basisklassen erstellt, allerdings konnten diese Klassen nicht als *abstract* definiert werden.

- Eine Klasse (XAML) mit WPF Technologie benötigt zwingend einen parameterlosen Konstruktor. Da unsere Klassen zwingend Parameter braucht, wurden die Klassen mit zwei Konstruktoren ausgestattet, wovon wir nur einen benutzen.

9.2.2 Spezifikation

Die Diplomarbeit wurde erschwert, da beim Auftraggeber das Projekt nicht in allen Punkten klar definiert war und unterschiedliche Vorstellungen vorhanden waren. Das bewirkte zusätzlichen Aufwand im Projektmanagement.

Ausserdem konnte der Auftraggeber in einigen Punkten sehr lange oder sogar bis zum Ende gar keine genauen Informationen liefern.

9.2.3 Integration Service (IS)

Da der IS nicht rechtzeitig zur Verfügung gestellt werden konnte, mussten wir einen eigenen, einfachen IS entwickeln, um unsere Anwendung testen zu können.

9.2.4 Software-Update

Zu diskutieren gab es auch bei den Software Update Funktionen. Im Pflichtenheft sind die Updates für Applikation und Konfiguration getrennt aufgeführt. Allerdings ist diese Trennung aufgrund des Lastenhefts entstanden, wobei mit der Konfiguration nicht unbedingt die Konfigurationsdatei der Terminal-Anwendung gemeint ist. Von daher machte es für uns nicht viel Sinn, das Update von Applikation und Konfiguration zu trennen.

9.3 Referenz-Anwendung "Brick Separierer"

Da der Ablauf des Arbeitsplatzes „Brick Separierer" nicht fertig definiert ist, wurde ein provisorischer Prozessablauf definiert und umgesetzt. Grösstwahrscheinlich ist dies nicht der letzte Stand. Die Umsetzung der Referenz-Anwendung hat sich jedoch sehr gut als Test für das Framework geeignet, was uns auf Fehler aufmerksam machte.

10 Ausblick

In diesem Kapitel werden verschiedene Verbesserungsvorschläge beschrieben. Ebenfalls sind die anstehenden Arbeiten aufgeführt, die nach der Diplomarbeit erledigt werden sollten.

10.1 Dokumentation

Um dem Entwickler einer Terminal-Anwendung die Arbeit zu erleichtern, ist es notwendig, ein Benutzerhandbuch zu verfassen. Die diversen Beispiele und die API Dokumentation ist zwar in vollem Umfang vorhanden, reicht aber nicht aus.

10.2 Hardware-Adapter

Zu Beginn war der Plan, die Hardware-Adapter nicht direkt in die Terminal-Anwendung (Framework) einzubinden, sondern Zusatzmodule für den IS (Integrations-Service) zu erstellen. Die Kommunikation sollte dann stets über den IS laufen. Allerdings hatten wir zu dieser Variante unsere Bedenken ausgesprochen, da überhaupt nicht gewährleistet war, ob der IS zum benötigten Zeitpunkt verfügbar sein wird. Im Einverständnis mit dem Auftraggeber haben wir uns auf die direkte Anbindung der Hardware-Adapter im Framework geeinigt. Sobald der IS in vollem Umfang verfügbar ist, sollten aber von den Hardware-Adaptern neue Module für den IS erstellt werden.

Gegebenenfalls muss das Framework an die neue Situation angepasst werden, sofern die angeschlossene Hardware über den IS verfügbar sein soll.

10.2.1 Geräte mit Ein-/Ausgabe

Das Package *Adapter* im Framework unterstützt nur Eingabe- oder Ausgabegeräte. Geräte mit bidirektionaler Kommunikation werden zurzeit nicht unterstützt.

Falls benötigt, muss eine weitere Klasse mit dem Inhalt der beiden Data Klassen (*ADataInput*, *ADataOutput*) erstellt werden, um Geräte mit bidirektionaler Kommunikation zu unterstützen.

10.2.2 Label-Drucker

Da der Laber-Drucker nicht rechtzeitig zur Verfügung gestellt werden konnte, muss der Hardware-Adapter für den Drucker angepasst werden.

10.2.3 Barcode-Leser

Der Hardware-Adapter für den Barcode-Leser konnte ohne offene Punkte realisiert werden.

10.3 Framework

Aufgrund des fehlenden IS wurden die benötigten Funktionen über einen Dummy-IS realisiert. Diese Schnittstelle ist beim Wechsel zum endgültigen IS anzupassen.

10.4 Referenz-Anwendung “Brick Separierer”

Da der Ablauf zum jetzigen Zeitpunkt nicht zu endgültig definiert ist, werden grösstwahrscheinlich Änderungen vorgenommen werden müssen.

11 Verzeichnisse

11.1 Abbildungsverzeichnis

Abbildung 1: Gesamtprozessablauf 7
Abbildung 2: Mindmap Gesamtfunktionalität 13
Abbildung 3: Komponentenübersicht 14
Abbildung 4: Anwendungsfallmodell für den Label-Drucker 15
Abbildung 5: Fachklassendiagramm Label-Drucker 17
Abbildung 6: Anwendungsfallmodell für den Barcode-Leser 19
Abbildung 7: Fachklassendiagramm Barcode-Leser 21
Abbildung 8: Anwendungsfallmodell der Basisfunktionalität im Framework 23
Abbildung 9: Anwendungsfallmodell der Operatorfunktionalität im Framework 25
Abbildung 10: Fachklassendiagramm Framework 30
Abbildung 11: Anwendungsfallmodell für den Brick Separierer 32
Abbildung 12: Ablauf Brick Separierer 33
Abbildung 13: Paketaufteilung Framework 37
Abbildung 14: GUI Grundelemente Framework 47
Abbildung 15: Virtuelle Tastatur (Numpad) 48
Abbildung 16: Bild Eingabe Auftrags-ID 49
Abbildung 17: Bild Übersicht Bricks 50
Abbildung 18: Bild Messung eines Bricks 51
Abbildung 19: Klassendiagramm Label Drucker 99
Abbildung 20: Klassendiagramm Barcode-Leser 100
Abbildung 21: Klassendiagramm Framework, Teil 1 101
Abbildung 22: Klassendiagramm Framework, Teil 2 102
Abbildung 23: Klassendiagramm Adapter 103
Abbildung 24: Klassendiagramm Communication 104
Abbildung 25: Klassendiagramm Configuration, Teil 1 105
Abbildung 26: Klassendiagramm Configuration, Teil 2 106
Abbildung 27: Klassendiagramm Configuration, Teil 3 107
Abbildung 28: Klassendiagramm GUI 108
Abbildung 29: Klassendiagramm Referenz-Anwendung „Brick Separierer“ 109

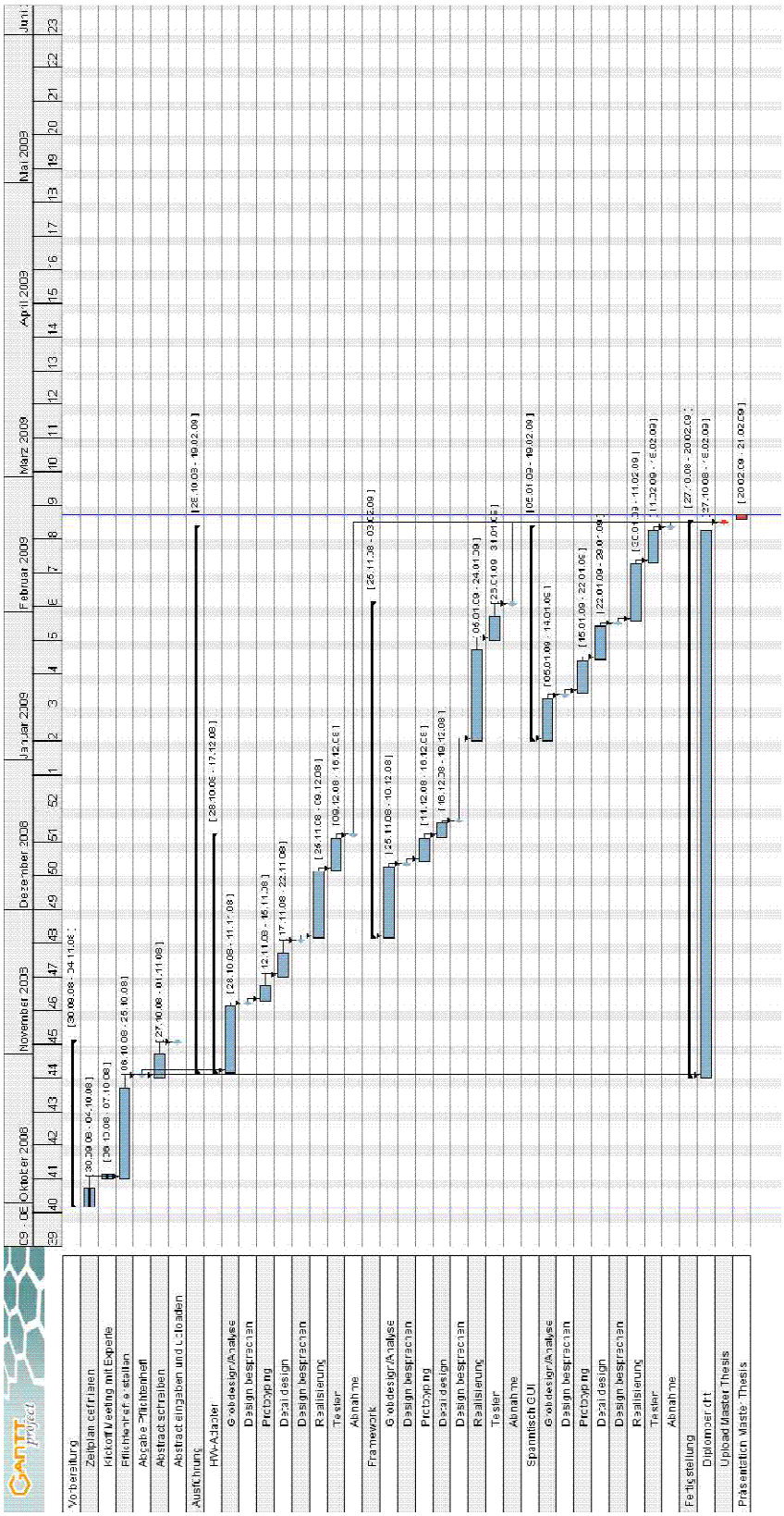
11.2 Tabellenverzeichnis

Tabelle 1: Dokumentversionen 6
Tabelle 2: Projektspezifisches Glossar 8
Tabelle 3: Anforderungsspezifisches Glossar 9
Tabelle 4: Projektspezifische Referenzen 10
Tabelle 5: Eingesetzte Werkzeuge 12
Tabelle 6: Synthese Label-Drucker 16
Tabelle 7: Synthese Barcode-Leser 20
Tabelle 8: Synthese Framework 26
Tabelle 9: Anwendungsfall-Beschreibung *Label-Drucker konfigurieren* 62
Tabelle 10: Anwendungsfall-Beschreibung *Vorlage herunterladen* 62
Tabelle 11: Anwendungsfall-Beschreibung *Vorlage speichern* 63
Tabelle 12: Anwendungsfall-Beschreibung *Vorlageninhalt interpretieren* 63
Tabelle 13: Anwendungsfall-Beschreibung *Label drucken* 64
Tabelle 14: Anwendungsfall-Beschreibung *Barcode-Leser konfigurieren* 65
Tabelle 15: Anwendungsfall-Beschreibung *Barcode lesen* 65
Tabelle 16: Anwendungsfall-Beschreibung *HW-Adapter hinzufügen* 66
Tabelle 17: Anwendungsfall-Beschreibung *Menü bearbeiten* 66
Tabelle 18: Anwendungsfall-Beschreibung *Fensterinhalt bearbeiten* 67
Tabelle 19: Anwendungsfall-Beschreibung *Fenster hinzufügen* 67
Tabelle 20: Anwendungsfall-Beschreibung *Sprache hinzufügen* 68
Tabelle 21: Anwendungsfall-Beschreibung *Bilder hinzufügen* 68
Tabelle 22: Anwendungsfall-Beschreibung *Barcodetyp aktivieren oder deaktivieren* 69
Tabelle 23: Anwendungsfall-Beschreibung *Fenster aktivieren oder deaktivieren* 69
Tabelle 24: Anwendungsfall-Beschreibung *Hilfsmittel aktivieren oder deaktivieren* 70
Tabelle 25: Anwendungsfall-Beschreibung *URL speichern* 70
Tabelle 26: Anwendungsfall-Beschreibung *Text-/XML-/HTML-Dokumente herunterladen* 71
Tabelle 27: Anwendungsfall-Beschreibung *Text-/XML-/HTML-Dokumente anzeigen* 71
Tabelle 28: Anwendungsfall-Beschreibung *Auftrags-ID senden* 71
Tabelle 29: Anwendungsfall-Beschreibung *ID-Antwort speichern* 72
Tabelle 30: Anwendungsfall-Beschreibung *Barcode einlesen* 72
Tabelle 31: Anwendungsfall-Beschreibung *Barcode aufsplitten* 73
Tabelle 32: Anwendungsfall-Beschreibung *Fehler anzeigen* 73
Tabelle 33: Anwendungsfall-Beschreibung *Produktionsstand speichern* 74
Tabelle 34: Anwendungsfall-Beschreibung *HW-Adapter laden* 74
Tabelle 35: Anwendungsfall-Beschreibung *Sprache ändern* 75
Tabelle 36: Anwendungsfall-Beschreibung *Sprachabhängige Bilder anzeigen* 75
Tabelle 37: Anwendungsfall-Beschreibung *Operator ID eingeben* 76
Tabelle 38: Anwendungsfall-Beschreibung *Operator-ID abbrechen* 76
Tabelle 39: Anwendungsfall-Beschreibung *Fenster öffnen* 77
Tabelle 40: Anwendungsfall-Beschreibung *Fenster schliessen* 78
Tabelle 41: Anwendungsfall-Beschreibung *Berechnung eingeben* 79
Tabelle 42: Anwendungsfall-Beschreibung *Software Update installieren* 79
Tabelle 43: Anwendungsfall-Beschreibung *Konfigurations Update installieren* 80
Tabelle 44: Anwendungsfall-Beschreibung *Produktionsstand ändern* 80
Tabelle 45: Anwendungsfall-Beschreibung *Auftrags-ID eingeben* 81
Tabelle 46: Anwendungsfall-Beschreibung *Bricks anzeigen* 81
Tabelle 47: Anwendungsfall-Beschreibung *Pass/Fail Eingabe* 82
Tabelle 48: Anwendungsfall-Beschreibung *Bricks messen* 82

Tabelle 49: Anwendungsfall-Beschreibung *Brick abschliessen* 83
Tabelle 50: Anwendungsfall-Beschreibung *Auftrag abschliessen* 83
Tabelle 51: Anwendungsfall-Beschreibung *Etikette ausdrucken* 83
Tabelle 52: Test Szenarien Label Drucker konfigurieren 84
Tabelle 53: Test Szenarien Vorlage herunterladen 85
Tabelle 54: Test Szenarien Vorlage speichern 85
Tabelle 55: Test Szenarien Vorlageninhalt interpretieren 86
Tabelle 56: Test Szenarien Label drucken 86
Tabelle 57: Test Szenarien Barcode-Leser konfigurieren 87
Tabelle 58: Test Szenarien Barcode lesen 87
Tabelle 59: Test Szenarien HW-Adapter hinzufügen 88
Tabelle 60: Test Szenarien Menü bearbeiten 88
Tabelle 61: Test Szenarien Fensterinhalt erstellen 89
Tabelle 62: Test Szenarien Fenster hinzufügen 89
Tabelle 63: Test Szenarien Sprache hinzufügen 89
Tabelle 64: Test Szenarien Bilder hinzufügen 89
Tabelle 65: Test Szenarien Barcode de-aktivieren 90
Tabelle 66: Test Szenarien Fenster de-aktivieren 90
Tabelle 67: Test Szenarien Hilfsmittel de-aktivieren 90
Tabelle 68: Test Szenarien Url speichern 90
Tabelle 69: TestSzenarien Dokumente herunterladen 90
Tabelle 70: Test Szenarien Dokumente anzeigen 91
Tabelle 71: Test Szenarien Auftrags-ID senden 91
Tabelle 72: Test Szenarien ID-Antwort speichern 91
Tabelle 73: Test Szenarien Barcode einlesen 92
Tabelle 74: Test Szenarien Barcode aufsplitten 92
Tabelle 75: Test Szenarien Status speichern 93
Tabelle 76: Test Szenarien HW-Adapter laden 93
Tabelle 77: Test Szenarien Sprache ändern 94
Tabelle 78: Test Szenarien Sprachabhängige Bilder anzeigen 94
Tabelle 79: Test Szenarien Operator-ID eingeben 94
Tabelle 80: Test Szenarien Operator-ID abbrechen 95
Tabelle 81: Test Szenarien Fenster öffnen 95
Tabelle 82: Test Szenarien Fenster schliessen 97
Tabelle 83: Test Szenarien Berechnung eingeben 97
Tabelle 84: Test Szenarien SW-Update installieren 97
Tabelle 85: Test Szenarien Konfig-Update installieren 98
Tabelle 86: Test Szenarien Produktionsstand ändern 98

12 Anhang

12.1 Projektplan



12.2 Anwendungsfall-Beschreibungen

12.2.1 Hardware-Adapter

12.2.1.1 Label Drucker

Tabelle 9: Anwendungsfall-Beschreibung *Label-Drucker konfigurieren*

Eigenschaft	Beschreibung
Name	Label-Drucker konfigurieren
Kurzbeschreibung	Die Hardware-Schnittstelle zum Label-Drucker mit den Parametern aus der entsprechenden Konfigurationsdatei konfigurieren und den Label-Drucker mit den Parametern aus der entsprechenden Konfigurationsdatei konfigurieren
Auslöser	Start Anwendung
Ergebnis	Hardware-Schnittstelle zum Label-Drucker richtig konfiguriert und Label-Drucker richtig konfiguriert
Akteur	System
Vorbedingungen	Label-Drucker angeschlossen Konfigurationsdateien vorhanden
Nachbedingungen	Label-Drucker einsatzbereit
Ablauf	1. Konfigurationsdatei Schnittstelle öffnen 2. Parameter lesen 3. Hardware-Schnittstelle konfigurieren 4. Konfigurationsdatei schliessen 5. Konfigurationsdatei Drucker öffnen 6. Parameter lesen 7. Drucker konfigurieren 8. Konfigurationsdatei schliessen 9. Konfigurationserfolg weiterleiten
Variationen	
Ausnahmen, Fehler	Schnittstelle besetzt Falsche Parameterwerte in der Konfigurationsdatei Konfigurationsdatei nicht vorhanden

Tabelle 10: Anwendungsfall-Beschreibung *Vorlage herunterladen*

Eigenschaft	Beschreibung
Name	Vorlage herunterladen
Kurzbeschreibung	Drucklayout-Vorlage vom PMS herunterladen
Auslöser	Neuer Auftrag eröffnet
Ergebnis	Drucklayout-Vorlage gespeichert
Akteur	System
Vorbedingungen	URL vorhanden
Nachbedingungen	
Ablauf	
Variationen	
Ausnahmen, Fehler	URL nicht vorhanden URL ungültig PMS Web-Server nicht erreichbar

Tabelle 11: Anwendungsfall-Beschreibung *Vorlage speichern*

Eigenschaft	Beschreibung
Name	Vorlage speichern
Kurzbeschreibung	Vom PMS heruntergeladene Drucklayout-Vorlage lokal speichern
Auslöser	Drucklayout-Vorlage heruntergeladen
Ergebnis	Drucklayout-Vorlage lokal gespeichert
Akteur	System
Vorbedingungen	Drucklayout-Vorlage heruntergeladen
Nachbedingungen	Datei gespeichert Datei am richtigen Ort (Ordner) und mit richtigem Dateinamen versehen
Ablauf	
Variationen	
Ausnahmen, Fehler	Keine Schreibrechte, um Datei lokal zu speichern Datei vorhanden

Tabelle 12: Anwendungsfall-Beschreibung *Vorlageninhalt interpretieren*

Eigenschaft	Beschreibung
Name	Vorlageninhalt interpretieren
Kurzbeschreibung	Den Inhalt der Vorlage interpretieren, das heisst, die Escape-Sequenzen mit gültigen Werten ersetzen und einen Druckinhalt erzeugen
Auslöser	Label drucken
Ergebnis	Druckinhalt vorbereitet
Akteur	System
Vorbedingungen	Vorlagendatei lokal vorhanden
Nachbedingungen	Druckinhalt im Arbeitspeicher vorhanden
Ablauf	1. Drucklayout-Vorlagendatei öffnen 2. Druckinhalt im Arbeitsspeicher vorbereiten 3. Inhalt der Datei lesen und parsen 4. Escape-Sequenzen interpretieren und durch gültige Werte ersetzen 5. Vorlagendatei schliessen
Variationen	
Ausnahmen, Fehler	Vorlagendatei nicht vorhanden Keine Leserechte zum Lesen der Vorlagendatei Fehler in Vorlagendatei

Tabelle 13: Anwendungsfall-Beschreibung *Label drucken*

Eigenschaft	Beschreibung
Name	Label drucken
Kurzbeschreibung	Label ausdrucken, indem der erzeugte Druckinhalt an den Drucker gesendet wird
Auslöser	Druckinhalt vorbereitet
Ergebnis	Label ausgedruckt
Akteur	System
Vorbedingungen	Druckinhalt vorbereitet
Nachbedingungen	Keine Fehlermeldung vom Drucker Label vom Drucker entnommen
Ablauf	1. Hardware-Schnittstelle öffnen 2. Druckinhalt an Drucker senden 3. Druckende vom Drucker abwarten 4. Hardware-Schnittstelle schliessen
Variationen	Bei Fehler wird kein Etikett ausgedruckt
Ausnahmen, Fehler	Hardware-Schnittstelle kann nicht geöffnet werden Drucker nicht bereit Drucker ist im Fehlerzustand (z.B. keine Etiketten, andere Fehler) Fehler während des Druckvorgangs

12.2.1.2 Barcode-Leser

Tabelle 14: Anwendungsfall-Beschreibung *Barcode-Leser konfigurieren*

Eigenschaft	Beschreibung
Name	Barcode-Leser konfigurieren
Kurzbeschreibung	Die Hardware-Schnittstelle zum Barcode-Leser mit den Parametern aus der entsprechenden Konfigurationsdatei konfigurieren
Auslöser	Start Anwendung
Ergebnis	Hardware-Schnittstelle zum Barcode-Leser richtig konfiguriert
Akteur	System
Vorbedingungen	Barcode-Leser angeschlossen Konfigurationsdatei vorhanden
Nachbedingungen	Barcode-Leser einsatzbereit
Ablauf	1. Konfigurationsdatei öffnen 2. Parameter lesen 3. Hardware-Schnittstelle konfigurieren 4. Konfigurationsdatei schliessen 5. Konfigurationserfolg weiterleiten
Variationen	
Ausnahmen, Fehler	Schnittstelle besetzt Falsche Parameterwerte in der Konfigurationsdatei Konfigurationsdatei nicht vorhanden

Tabelle 15: Anwendungsfall-Beschreibung *Barcode lesen*

Eigenschaft	Beschreibung
Name	Barcode lesen
Kurzbeschreibung	Ein Barcode mit dem Barcode-Leser einlesen
Auslöser	Barcode muss erfasst werden
Ergebnis	Barcode eingelesen
Akteur	Benutzer
Vorbedingungen	Barcode-Leser einsatzbereit
Nachbedingungen	
Ablauf	1. Hardware-Schnittstelle öffnen 2. Meldung lesen 3. Hardware-Schnittstelle schliessen 4. Meldung weiterleiten
Variationen	
Ausnahmen, Fehler	Hardware-Schnittstelle kann nicht geöffnet werden Leser nicht bereit

12.2.2 Framework

12.2.2.1 Entwickler

Tabelle 16: Anwendungsfall-Beschreibung *HW-Adapter hinzufügen*

Eigenschaft	Beschreibung
Name	HW-Adapter hinzufügen
Kurzbeschreibung	Einen Hardware-Adapter zum Framework hinzufügen
Auslöser	Für das Erstellen einer neuer Terminalanwendung wird ein neuer HW-Adapter wird benötigt
Ergebnis	Der HW-Adapter ist in der Konfigurationsdatei eingetragen
Akteur	Entwickler
Vorbedingungen	
Nachbedingungen	Der HW-Adapter ist bereit um dynamisch eingefügt zu werden
Ablauf	1. In der Konfigurationsdatei wird der Pfad und Namen der HW-Konfigurationsdatei und Bibliothek eintragen 2. Der Entwickler hat dafür zu sorgen, dass die Konfigurationsdatei für den HW-Adapter und die dazugehörige Bibliothek zur Verfügung stehen.
Variationen	
Ausnahmen, Fehler	Keine Bibliotheke oder Konfigurationsdatei für den HW-Adapter am vorgehsehenen Ort vorhanden Ungültiger oder kein Pfad in der Konfigurationsdatei zum HW-Adapter vorhanden

Tabelle 17: Anwendungsfall-Beschreibung *Menü bearbeiten*

Eigenschaft	Beschreibung
Name	Menü bearbeiten
Kurzbeschreibung	Die Struktur des Menüs bearbeiten, anhand der zu erstellenden Terminalanwendung
Auslöser	Ein neues Terminal wird erstellt
Ergebnis	Die Menüstruktur der Terminalanwendung stimmt mit der zu erstellende Anwendung (pro Screen) überein
Akteur	Entwickler
Vorbedingungen	
Nachbedingungen	
Ablauf	1. Das Menü editieren
Variationen	Menüpunkt umplatzieren Menüpunkt löschen Menüpunkt einfügen
Ausnahmen, Fehler	Das falsche Menü ist für die falschen Seiten Das Menüpunkt verweist auf eine nicht existierende Seite

Tabelle 18: Anwendungsfall-Beschreibung *Fensterinhalt bearbeiten*

Eigenschaft	Beschreibung
Name	Fensterinhalt bearbeiten
Kurzbeschreibung	Bei einer neuen Terminalanwendung wird der Inhalt einer Seite bearbeitet
Auslöser	Ein neues Terminal wird erstellt
Ergebnis	Der Inhalt einer neuen Seite ist erstellt worden oder der Inhalt ist bearbeitet worden
Akteur	Entwickler
Vorbedingungen	Die zu bearbeitende Seite besteht bereits
Nachbedingungen	Der Inhalt der Seite ist geändert
Ablauf	1. Den Inhalt eines Fensters bearbeiten
Variationen	
Ausnahmen, Fehler	Anhand des Platzbedarfs des Fensters, wird das Layout nicht richtig angezeigt

Tabelle 19: Anwendungsfall-Beschreibung *Fenster hinzufügen*

Eigenschaft	Beschreibung
Name	Fenster hinzufügen
Kurzbeschreibung	Ein neues Fenster wird der Terminalanwendung hinzugefügt
Auslöser	Eine weitere Seite muss ergänzt werden
Ergebnis	Eine neue Seite ist erstellt worden
Akteur	Entwickler
Vorbedingungen	
Nachbedingungen	Eine neue Seite besteht
Ablauf	1. Neue Seite hinzufügen
Variationen	
Ausnahmen, Fehler	Bsp. Anzahl Seiten im Register ist voll ausgeschöpft und es kann keine weitere Seite mehr eingefügt werden.

Tabelle 20: Anwendungsfall-Beschreibung *Sprache hinzufügen*

Eigenschaft	Beschreibung
Name	Sprache hinzufügen
Kurzbeschreibung	Bei einer vorhandenen Terminalanwendung kann die Sprachbibliothek erweitert werden. Dies kann auch für nicht lateinische Sprachen vorkommen (Bsp. Chinesisch)
Auslöser	Eine weitere Sprache muss ergänzt werden
Ergebnis	Die Sprache für die Terminalanwendung kann anhand der Sprachbibliothek erweitert werden
Akteur	Entwickler
Vorbedingungen	Sprachbibliothek ist korrekt vorhanden
Nachbedingungen	
Ablauf	1. Die Sprachbibliothek wird in den dafür vorgesehenen Ordner kopiert 2. Das Auswahlelement für die Sprachauswahl muss erstellt werden 3. Die Konfigurationsdatei muss für die Sprache erweitert werden
Variationen	
Ausnahmen, Fehler	Falsche Texte für die falschen Variablen Zeichensatz wird nicht unterstützt Die SprachBibliotheke kann nicht eingebunden werden Zu lange Textnamen, damit ein Textfeld den Text nicht vollständig anzeigt Das Auswahlelement für die Auswahl der Sprache existiert nicht Sprachbibliothek ist nicht komplett

Tabelle 21: Anwendungsfall-Beschreibung *Bilder hinzufügen*

Eigenschaft	Beschreibung
Name	Bilder hinzufügen
Kurzbeschreibung	Bei einer zu erstellenden oder bestehenden Terminalanwendung können zusätzliche Bilder hinzugefügt werden
Auslöser	Weitere Bilder müssen hinzugefügt werden
Ergebnis	Die Bilder werden korrekt und in der richtigen Reihenfolge angezeigt
Akteur	Entwickler
Vorbedingungen	Bilder sind richtig beschriftet Bilder haben die richtige Grösse
Nachbedingungen	
Ablauf	1. Bilder in den vorgesehenen Ordner einfügen
Variationen	
Ausnahmen, Fehler	Falsche Bildgrösse Falsche Bildreihenfolge (falsche Bildernummerierung)

Tabelle 22: Anwendungsfall-Beschreibung *Barcodetyp aktivieren oder deaktivieren*

Eigenschaft	Beschreibung
Name	Barcodetyp aktivieren oder deaktivieren
Kurzbeschreibung	Jeder Barcodetyp kann mittels einer Konfigurationsdatei aktiviert oder deaktiviert werden
Auslöser	Der Entwickler will einen Barcodetyp aktivieren oder deaktivieren
Ergebnis	Der Barcodetyp ist aktiv oder nicht aktiv
Akteur	Entwickler
Vorbedingungen	
Nachbedingungen	
Ablauf	1. Konfigurationsdatei öffnen 2. Barcodetyp aktivieren oder deaktivieren 3. Konfigurationsdatei speichern und schliessen
Variationen	
Ausnahmen, Fehler	Konfigurationsdatei kann nicht geöffnet werden Kein Schreibrecht in der Konfigurationsdatei Barcodetyp ist in der Konfigurationsdatei nicht vorhanden

Tabelle 23: Anwendungsfall-Beschreibung *Fenster aktivieren oder deaktivieren*

Eigenschaft	Beschreibung
Name	Fenster aktivieren oder deaktivieren
Kurzbeschreibung	Jedes Fenster kann mittels einer Konfigurationsdatei aktiviert oder deaktiviert werden
Auslöser	Der Entwickler will ein Fenster aktivieren oder deaktivieren
Ergebnis	Das Fenster ist aktiv oder nicht aktiv
Akteur	Entwickler
Vorbedingungen	
Nachbedingungen	
Ablauf	1. Konfigurationsdatei öffnen 2. Fenster aktivieren oder deaktivieren 3. Konfigurationsdatei speichern und schliessen
Variationen	
Ausnahmen, Fehler	Konfigurationsdatei kann nicht geöffnet werden Kein Schreibrecht in der Konfigurationsdatei Das Fenster ist in der Konfigurationsdatei nicht vorhanden

Tabelle 24: Anwendungsfall-Beschreibung *Hilfsmittel aktivieren oder deaktivieren*

Eigenschaft	Beschreibung
Name	Hilfsmittel aktivieren oder deaktivieren
Kurzbeschreibung	Jedes Hilfsmittel kann mittels einer Konfigurationsdatei aktiviert oder deaktiviert werden
Auslöser	Der Entwickler will ein Hilfsmittel aktivieren oder deaktivieren
Ergebnis	Das Hilfsmittel ist aktiv oder nicht aktiv
Akteur	Entwickler
Vorbedingungen	
Nachbedingungen	
Ablauf	1. Konfigurationsdatei öffnen 2. Hilfsmittel aktivieren oder deaktivieren 3. Konfigurationsdatei speichern und schliessen
Variationen	
Ausnahmen, Fehler	Konfigurationsdatei kann nicht geöffnet werden Kein Schreibrecht in der Konfigurationsdatei Hilfsmittel ist in der Konfigurationsdatei nicht vorhanden

12.2.2.2 System

Tabelle 25: Anwendungsfall-Beschreibung *URL speichern*

Eigenschaft	Beschreibung
Name	URL speichern
Kurzbeschreibung	Die vom AS erhaltene URL speichern
Auslöser	URL von AS erhalten
Ergebnis	URL gespeichert
Akteur	System
Vorbedingungen	
Nachbedingungen	
Ablauf	1. Die URL im System abspeichern
Variationen	
Ausnahmen, Fehler	

Tabelle 26: Anwendungsfall-Beschreibung *Text-/XML-/HTML-Dokumente herunterladen*

Eigenschaft	Beschreibung
Name	Text-/XML-/HTML-Dokumente herunterladen
Kurzbeschreibung	Text-/XML-/HTML-Dokumente vom PMS herunterladen
Auslöser	Sobald Dokumente vom Ablauf benutzt werden müssen
Ergebnis	Text-/XML-/HTML-Dokumente gespeichert
Akteur	System
Vorbedingungen	URL vorhanden
Nachbedingungen	
Ablauf	
Variationen	
Ausnahmen, Fehler	URL nicht vorhanden PMS Web-Server nicht erreichbar

Tabelle 27: Anwendungsfall-Beschreibung *Text-/XML-/HTML-Dokumente anzeigen*

Eigenschaft	Beschreibung
Name	Text-/XML-/HTML-Dokumente anzeigen
Kurzbeschreibung	Gespeicherte Text-/XML-/HTML-Dokumente anzeigen
Auslöser	Sobald Dokumente vom Ablauf benutzt werden müssen
Ergebnis	Text-/XML-/HTML-Dokumente angezeigt
Akteur	System
Vorbedingungen	Text-/XML-/HTML-Dokumente heruntergeladen
Nachbedingungen	
Ablauf	
Variationen	
Ausnahmen, Fehler	Text-/XML-/HTML-Dokumente nicht vorhanden

Tabelle 28: Anwendungsfall-Beschreibung *Auftrags-ID senden*

Eigenschaft	Beschreibung
Name	Auftrags-ID senden
Kurzbeschreibung	Die Auftrags-ID ans PMS senden
Auslöser	Neuer Auftrag eröffnen
Ergebnis	Die ID ist gesendet
Akteur	System
Vorbedingungen	Kein offener Auftrag
Nachbedingungen	
Ablauf	1. Auftrags-ID eingeben 2. ID ans PMS senden
Variationen	
Ausnahmen, Fehler	Falsche oder ungültige ID AS nicht erreichbar

Tabelle 29: Anwendungsfall-Beschreibung *ID-Antwort speichern*

Eigenschaft	Beschreibung
Name	ID-Antwort speichern
Kurzbeschreibung	Nachdem die Auftragsdaten ans PMS gesendet worden sind, kommt vom PMS eine Antwort mit den nötigen Daten
Auslöser	ID senden (neuer Auftrag eröffnen)
Ergebnis	Alle benötigten Daten zu einem Auftrag sind lokal gespeichert
Akteur	System
Vorbedingungen	Auftrag-ID muss ans PMS gesendet werden
Nachbedingungen	
Ablauf	1. Die Daten zum Auftrag werden vom PMS als Antwort gespeichert
Variationen	
Ausnahmen, Fehler	Keine oder nicht alle Auftragsdaten wurden zurückgegeben
Regeln	
Invarianten	
Inkludierte AF	Auftrags-ID senden

Tabelle 30: Anwendungsfall-Beschreibung *Barcode einlesen*

Eigenschaft	Beschreibung
Name	Barcode einlesen
Kurzbeschreibung	Der Barcode wird über den HW-Adapter eingelesen
Auslöser	Eine Identifikationsnummer wird benötigt
Ergebnis	Der Wert eines Barcodes ist gespeichert
Akteur	System
Vorbedingungen	Barcode wurde vom HW-Adapter richtig eingelesen
Nachbedingungen	
Ablauf	1. System erhält eine Meldung vom HW-Adapter, dass ein Barcode vorhanden ist 2. Das System liest den Barcode ein 3. Der Wert des Barcodes wird im Arbeitsspeicher gespeichert
Variationen	
Ausnahmen, Fehler	Kein oder ein ungültiger Barcode vorhanden

Tabelle 31: Anwendungsfall-Beschreibung *Barcode aufsplitten*

Eigenschaft	Beschreibung
Name	Barcode aufsplitten
Kurzbeschreibung	Der eingelesene Wert des Barcodes wird dem dazugehörigen Typ zugeordnet
Auslöser	Barcode wurde eingelesen
Ergebnis	Der Wert eines Barcodes ist dem richtigen Barcode-Typ zugeordnet
Akteur	System
Vorbedingungen	Barcode wurde vom HW-Adapter richtig eingelesen
Nachbedingungen	
Ablauf	1. Das System liest den Wert des Barcodes aus dem Speicher 2. Das Framework identifiziert den Barcode mit dem jeweiligen Typ (Brick/Ingot/…) 3. Der Wert des Barcodes wird ins Feld eingetragen, wenn der Typ dazu übereinstimmt 4. Nach dem Drücken der Enter-Taste wird der Barcode übernommen
Variationen	
Ausnahmen, Fehler	Barcode-Identifikation findet den Typ nicht Falscher Barcode-Typ

Tabelle 32: Anwendungsfall-Beschreibung *Fehler anzeigen*

Eigenschaft	Beschreibung
Name	Fehler anzeigen
Kurzbeschreibung	Wenn der Barcode ungültig ist, wird ein Fehler angezeigt
Auslöser	Keine oder ungültiger Barcode
Ergebnis	Ein Dialog mit einer Fehlermeldung wird angezeigt
Akteur	System
Vorbedingungen	Der Barcode wurde aufgesplittet und es war kein oder ein ungültiger Wert vorhanden
Nachbedingungen	
Ablauf	1. Eine Meldung erscheint, dass kein oder ein ungültiger Wert des Barcodes vorhanden ist
Variationen	
Ausnahmen, Fehler	Dialogfenster wird nicht angezeigt
Regeln	
Invarianten	
Inkludierte AF	Barcode aufsplitten

Tabelle 33: Anwendungsfall-Beschreibung *Produktionsstand speichern*

Eigenschaft	Beschreibung
Name	Produktionsstand speichern
Kurzbeschreibung	Der Status des Produktionsstands wird gespeichert und ans PMS gesendet
Auslöser	Der Status wird geändert
Ergebnis	Der Status ist gespeichert
Akteur	System
Vorbedingungen	Der Status des Produktionsstands hat geändert
Nachbedingungen	
Ablauf	1. Status ändert 2. Status wird gespeichert 3. Status wird ans PMS gesendet
Variationen	
Ausnahmen, Fehler	PMS ist nicht ansprechbar Das Speichern ist nicht möglich

Tabelle 34: Anwendungsfall-Beschreibung *HW-Adapter laden*

Eigenschaft	Beschreibung
Name	HW-Adapter laden
Kurzbeschreibung	Beim Starten der Terminalanwendung werden die konfigurierten HW-Adapter geladen
Auslöser	
Ergebnis	Alle HW-Adapter sind im System integriert und können die Aufgaben wahrnehmen
Akteur	System
Vorbedingungen	Die Bibliothek muss vorhanden sein Die dazugehörige Konfigurationsdatei muss vorhanden sein
Nachbedingungen	
Ablauf	1. HW-Adapter (Bibliothek) dynamisch laden 2. Die HW gegebenenfalls konfigurieren
Variationen	
Ausnahmen, Fehler	HW-nicht angeschlossen

12.2.2.3 Operator

Tabelle 35: Anwendungsfall-Beschreibung *Sprache ändern*

Eigenschaft	Beschreibung
Name	Sprache ändern
Kurzbeschreibung	Die Sprache wird anhand der ausgewählten Sprache geändert und angezeigt
Auslöser	Der Bediener ändert die Sprache
Ergebnis	Die Terminalanwendung wird in der gewünschten Sprache angezeigt
Akteur	Operator
Vorbedingungen	Die gewünschte Sprache ist als Sprachtabelle vorhanden und installiert Die Sprachumstellung kann umgestellt werden
Nachbedingungen	
Ablauf	1. Beim Starten wird die in der Konfigurationsdatei eingestellte Sprache angezeigt 2. Der Benutzer wählt die gewünschte Sprache, indem er auf den jeweiligen Button drückt 3. Die Anwendung erscheint in der gewünschten Sprache 4. Die gewählte Sprache wird in der Konfigurationsdatei gesetzt
Variationen	
Ausnahmen, Fehler	Die Sprache stellt nicht um Die Bibliothek ist nicht vorhanden Die Bibliothek ist unvollständig Die gewünschte Sprache kann nicht umgestellt werden

Tabelle 36: Anwendungsfall-Beschreibung *Sprachabhängige Bilder anzeigen*

Eigenschaft	Beschreibung
Name	Sprachabhängige Bilder anzeigen
Kurzbeschreibung	Anhand der gewählten Sprache, die sprachabhängigen Bilder anzeigen
Auslöser	Sprache ändern (Benutzer)
Ergebnis	Die sprachabhängigen Bilder werden angezeigt
Akteur	Operator
Vorbedingungen	Sprache wurde geändert oder beim Starten wird in der Konfigurationsdatei die gewählte Sprache initialisiert
Nachbedingungen	Wenn ein Fenster mit Bildinhalt geöffnet wird, dann werden anhand der ausgewählten Sprache die sprachabhängigen Bilder angezeigt
Ablauf	Wenn ein Fenster mit Bildinhalt geöffnet wird, dann werden anhand der ausgewählten Sprache die sprachabhängigen Bilder angezeigt
Variationen	
Ausnahmen, Fehler	Bilder in der entsprechenden Sprache ist nicht vorhanden
Regeln	
Invarianten	
Inkludierte AF	Sprache ändern

Tabelle 37: Anwendungsfall-Beschreibung *Operator ID eingeben*

Eigenschaft	Beschreibung
Name	Operator ID eingeben
Kurzbeschreibung	Der Operator gibt seine ID ein
Auslöser	Eine Anmeldung wird verlangt
Ergebnis	Der Operator ist angemeldet
Akteur	Operator
Vorbedingungen	Anmeldefenster offen Eingabefeld fokusiert Cursor auf Eingabefeld Möglichkeit zum Öffnen der virtuellen Tastatur vorhanden Optional: Barcode-Leser bereit
Nachbedingungen	
Ablauf	1. Cursor und Fokus auf entsprechendes Eingabefeld setzen 2. Virtuelle Tastatur öffnen nach Befehl "Öffnen der virtuellen Tastatur" 3. Identifikationsnummer über virtuelle Tastatur eingeben 4. Bestätigen mit Enter 5. Identifikationsnummer wird übernommen 6. Virtuelle Tastatur schliessen
Variationen	Barcode Leser: 1. Cursor und Fokus auf entsprechendes Eingabefeld setzen 2. Identifikationsnummer über Barcode-Leser einlesen 3. Automatische Bestätigung
Ausnahmen, Fehler	Keine oder eine ungültige ID vorhanden

Tabelle 38: Anwendungsfall-Beschreibung *Operator-ID abbrechen*

Eigenschaft	Beschreibung
Name	Operator-ID abbrechen
Kurzbeschreibung	Die Eingabe der Operator-ID wird abgebrochen
Auslöser	Der Operator will die Eingabe der Operator-ID abbrechen
Ergebnis	Die Operator-ID wird nicht übernommen
Akteur	Operator
Vorbedingungen	Der Operator versuchte vorher seine ID einzugeben
Nachbedingungen	
Ablauf	Abbrechen der Operator-ID
Variationen	
Ausnahmen, Fehler	

Tabelle 39: Anwendungsfall-Beschreibung *Fenster öffnen*

Eigenschaft	Beschreibung
Name	Fenster öffnen
Kurzbeschreibung	Das Fenster mit dem Dokument/Hilfsmittel/Beschreibung wird geöffnet und angezeigt
Auslöser	Aufruf des Fensters über das Menü
Ergebnis	Der Fenster wird angezeigt
Akteur	Operator
Vorbedingungen	Das Fenster mit dem Dokument/Hilfsmittel/Beschreibung ist vorhanden Der Aufruf des Fensters ist in mindestens einem Menüpunkt hinterlegt Das Fenster ist aktiviert
Nachbedingungen	
Ablauf	1. Das Fenster mit dem Dokument/Hilfsmittel/Beschreibung wird geöffnet 2. Das Fenster mit dem Dokument/Hilfsmittel/Beschreibung wird angezeigt
Variationen	
Ausnahmen, Fehler	Die Datei mit dem Dokument/Beschreibung ist nicht vorhanden Die Datei mit dem Dokument/Beschreibung ist lese-geschützt
Regeln	
Invarianten	
Inkludierte AF	
Bemerkungen	**Dieser Anwendungsfall gilt für:** Arbeitsbeschreibungshilfe Auftragsinformationen Informationen zu nächsten Aufträgen Wartungsinformationen Bildschirm-Reinigerhilfe Bilder-Assistent Test-Bedingungshilfe Rechner Kalender Software-Version

Tabelle 40: Anwendungsfall-Beschreibung *Fenster schliessen*

Eigenschaft	Beschreibung
Name	Fenster schliessen
Kurzbeschreibung	Das Fenster mit dem Dokument/Hilfsmittel/Beschreibung etc wird geschlossen
Auslöser	Aufruf eines anderen Fensters über das Menü
Ergebnis	Das Fenster ist geschlossen
Akteur	Operator
Vorbedingungen	Das Fenster mit dem Dokument/Hilfsmittel/Beschreibung etc wird angezeigt
Nachbedingungen	
Ablauf	1. Das Fenster mit dem Dokument/Hilfsmittel/Beschreibung etc wird nicht mehr angezeigt 2. Das Fenster mit dem Dokument/Hilfsmittel/Beschreibung etc wird geschlossen
Variationen	
Ausnahmen, Fehler	Ein Dialog ist nicht als modal offen
Regeln	
Invarianten	
Inkludierte AF	
Bemerkungen	**Dieser Anwendungsfall gilt für:** Arbeitsbeschreibungshilfe Auftragsinformationen Informationen zu nächsten Aufträgen Wartungsinformationen Bildschirm-Reinigerhilfe Bilder-Assistent Test-Bedingungshilfe Rechner Kalender Software-Version

Tabelle 41: Anwendungsfall-Beschreibung *Berechnung eingeben*

Eigenschaft	Beschreibung
Name	Berechnung eingeben
Kurzbeschreibung	Es kann eine Rechnung in den Rechner eingegeben werden
Auslöser	Der Operator möchte eine Berechnung ausführen
Ergebnis	Das Ergebnis der Berechnung wird angezeigt
Akteur	Operator
Vorbedingungen	Der Rechner ist geöffnet
Nachbedingungen	Das Resultat der Berechnung wird angezeigt
Ablauf	1. Die erste Zahl wird eingegeben 2. Die Operation wird eingegeben 3. Die zweite Zahl wird eingegeben 4. Das "Gleichheits"-Zeichen wird gedrückt 5. Das Resultat wird angezeigt 6. Das Resultat kann wieder als erste Zahl benutzt werden
Variationen	
Ausnahmen, Fehler	Der Ablauf wird nicht eingehalten Falsche Konfiguration des Touch (virtuelle Tastatur wird nicht angezeigt)

Tabelle 42: Anwendungsfall-Beschreibung *Software Update installieren*

Eigenschaft	Beschreibung
Name	Software Update installieren
Kurzbeschreibung	Ein Software Update wird installiert
Auslöser	Der Operator möchte die neuste Software-Version
Ergebnis	Die neuste Software ist installiert
Akteur	Operator
Vorbedingungen	Ein Software Update ist vorhanden
Nachbedingungen	
Ablauf	1. Überprüfen ob eine neuere Version der Software auf dem PMS vorhanden ist 2. Software vom PMS herunterladen 3. Software installieren 4. Software neu starten
Variationen	
Ausnahmen, Fehler	Keine neue Software vorhanden Installation konnte nicht ausgeführt werden

Tabelle 43: Anwendungsfall-Beschreibung *Konfigurations Update installieren*

Eigenschaft	Beschreibung
Name	Konfigurations Update installieren
Kurzbeschreibung	Ein Konfigurations Update wird installiert
Auslöser	Der Operator möchte die neuste Version oder ein Hilfsmittel de- oder aktivieren
Ergebnis	Die neuste Konfiguration ist installiert
Akteur	Operator
Vorbedingungen	Ein Konfigurations Update ist vorhanden
Nachbedingungen	
Ablauf	1. Überprüfen ob eine neuere Version der Konfiguration auf dem PMS vorhanden ist 2. Konfiguration herunterladen 3. Konfiguration speichern 4. Konfiguration neu laden
Variationen	
Ausnahmen, Fehler	Keine neue Konfiguration vorhanden Installation konnte nicht ausgeführt werden

Tabelle 44: Anwendungsfall-Beschreibung *Produktionsstand ändern*

Eigenschaft	Beschreibung
Name	Produktionsstand ändern
Kurzbeschreibung	Der Status des Produktionsstands wird geändert
Auslöser	Der Produktionsstatus ändert sich oder wird vom Operator übersteuert
Ergebnis	Der Status ist geändert
Akteur	Operator oder System
Vorbedingungen	
Nachbedingungen	
Ablauf	1. Status ändert
Variationen	
Ausnahmen, Fehler	Änderung findet im System nicht statt.

12.2.3 Referenz-Anwendung „Brick Separierer“

Tabelle 45: Anwendungsfall-Beschreibung *Auftrags-ID eingeben*

Eigenschaft	Beschreibung
Name	Auftrags-ID eingeben
Kurzbeschreibung	Der Benutzer gibt die Auftrags-ID von Hand oder mittels Barcode Leser ein.
Auslöser	Ein Auftrag soll abgearbeitet werden
Ergebnis	Ein gültiger Auftrag soll bearbeitet werden können
Akteur	Bediener
Vorbedingungen	Die Anwendung ist gestartet
Nachbedingungen	Der Auftrag kann bearbeitet werden Die im Auftrag enthaltenen Bricks werden angezeigt
Ablauf	
Variationen	Eingabe der ID von Hand Eingabe der ID mittels Barcode Leser
Ausnahmen, Fehler	Falsche Auftrags-ID

Tabelle 46: Anwendungsfall-Beschreibung *Bricks anzeigen*

Eigenschaft	Beschreibung
Name	Bricks anzeigen
Kurzbeschreibung	Alle zu bearbeitenden Bricks werden angezeigt
Auslöser	Eine gültige Auftrags-ID wurde eingegeben und ein Rezept vom Server geladen
Ergebnis	Ein gültiger Auftrag soll bearbeitet werden können
Akteur	System
Vorbedingungen	Eine gültige Auftrags-ID wurde eingegeben und ein Rezept vom Server geladen
Nachbedingungen	Die Bricks werden angezeigt
Ablauf	
Variationen	
Ausnahmen, Fehler	Ungültiges Rezept

Tabelle 47: Anwendungsfall-Beschreibung *Pass/Fail Eingabe*

Eigenschaft	Beschreibung
Name	Pass/Fail Eingabe
Kurzbeschreibung	Der Benutzer gibt anhand einer visuellen Kontrolle ein Pass oder Fail für den angezeigten Brick
Auslöser	Ein Brick soll bearbeitet werden
Ergebnis	Die Anwendung zeigt dem Benutzer ob bei Pass eine Messung durchgeführt werden muss
Akteur	Bediener
Vorbedingungen	Ein Auftrag ist aktiv
Nachbedingungen	Bei Pass kann anhand des Rezepts eine Messung verlangt werden Der nexte Brick wird angezeigt Der Brick wird als "done" beim PMS gemeldet
Ablauf	1. Eingabe von Pass oder Fail
Variationen	Bei Pass kann vom Rezept her eine Messung verlangt werden Bei Fail wird der nächste Brick angezeigt
Ausnahmen, Fehler	

Tabelle 48: Anwendungsfall-Beschreibung *Bricks messen*

Eigenschaft	Beschreibung
Name	Bricks messen
Kurzbeschreibung	Der Benutzer vermisst die Bricks anhand der vorgegebenen Bedingungen
Auslöser	Ein Brick soll vermessen werden
Ergebnis	Die Messdaten werden in der Anwendung eingegeben und ans PMS weitergeleitet
Akteur	Bediener
Vorbedingungen	Der zu messende Brick muss vermessen werden
Nachbedingungen	Die Messdaten werden ans PMS gesendet Anhand der gemessenen Werte müssen nur ein Teil oder alle Bricks vermessen werden
Ablauf	1. Die Breite 1 wird gemessen und eingegeben 2. Die Breite 2 wird gemessen und eingegeben 3. Die Länge wird gemessen und eingegeben 4. Die Messung wird abgeschlossen
Variationen	
Ausnahmen, Fehler	
Regeln	Anhand des Rezeptes gibt es muss und soll Messungen. Ist eine Muss-Messung nicht inerhalb einer Toleranz müssen allle Bricks vermessen werden

Tabelle 49: Anwendungsfall-Beschreibung *Brick abschliessen*

Eigenschaft	Beschreibung
Name	Brick abschliessen
Kurzbeschreibung	Nach dem Bearbeiten der Bricks muss dem PMS mitgeteilt werden, dass der aktuelle Brick fertig bearbeitet wurde.
Auslöser	Der zu bearbeitende Brick ist fertig bearbeitet
Ergebnis	Der aktuelle Brick ist beim PMS als bearbeitet markiert
Akteur	Bediener
Vorbedingungen	Der zu bearbeitende Brick ist fertig bearbeitet
Nachbedingungen	Für den aktuellen Brick wird eine Etikette ausgedruckt
Ablauf	
Variationen	
Ausnahmen, Fehler	

Tabelle 50: Anwendungsfall-Beschreibung *Auftrag abschliessen*

Eigenschaft	Beschreibung
Name	Auftrag abschliessen
Kurzbeschreibung	Nach dem Bearbeiten sämtlicher Bricks muss dem PMS mitgeteilt werden, dass der aktuelle Auftrag fertiggestellt wurde.
Auslöser	Sämtliche Bricks sind fertig bearbeitet
Ergebnis	Das PMS hat den Status abgeschlossen für den aktuellen Auftrag
Akteur	Bediener
Vorbedingungen	Sämtliche Bricks sind fertig bearbeitet
Nachbedingungen	
Ablauf	
Variationen	
Ausnahmen, Fehler	

Tabelle 51: Anwendungsfall-Beschreibung *Etikette ausdrucken*

Eigenschaft	Beschreibung
Name	Etikette ausdrucken
Kurzbeschreibung	Nach dem Bearbeiten des aktuellen Bricks muss eine Etikette ausgedruckt werden
Auslöser	Sämtliche Bricks sind fertig bearbeitet
Ergebnis	Auf den Brick klebt eine Etikette
Akteur	System
Vorbedingungen	Sämtliche Bricks sind fertig bearbeitet Es ist ein Template zum Drucken der Etikette vorhanden
Nachbedingungen	
Ablauf	
Variationen	
Ausnahmen, Fehler	

12.3 Test Szenarien

12.3.1 Hardware-Adapter

12.3.1.1 Label Drucker

Tabelle 52: Test Szenarien Label Drucker konfigurieren

Nr.	Szenario	Beschreibung	Ausgangslage	Erwartetes Resultat	Erfolg
LDK1	Normal konfiguriert	Die Anwendung startet und die Hardware-Schnittstelle zum Label-Drucker wird mittels den Parameterwerten aus der Konfigurationsdatei konfiguriert	Die Schnittstelle ist bereit Die Konfigurationsdatei ist vorhanden Die Parameterwerte in der Konfigurationsdatei sind korrekt	Hardware-Schnittstelle konfiguriert Anwendung initialisiert weiter	ok
LDK2	Schnittstelle besetzt	Die Anwendung startet und versucht die Hardware-Schnittstelle zum Label-Drucker mittels den Parameterwerten aus der Konfigurationsdatei zu konfigurieren.	Die Schnittstelle ist nicht bereit Die Konfigurationsdatei ist vorhanden Die Parameterwerte in der Konfigurationsdatei sind korrekt	Fehlermeldung wird angezeigt Anwendung beendet	ok
LDK3	Falsche Parameterwerte in Konfigurationsdatei	Die Anwendung startet und versucht die Hardware-Schnittstelle zum Label-Drucker mittels den Parameterwerten aus der Konfigurationsdatei zu konfigurieren.	Die Schnittstelle ist bereit Die Konfigurationsdatei ist vorhanden Mindestens 1 Parameterwert in der Konfigurationsdatei ist falsch	Fehlermeldung wird angezeigt Anwendung beendet	ok
LDK4	Konfigurationsdatei fehlt	Die Anwendung startet und versucht die Hardware-Schnittstelle zum Label-Drucker mittels den Parameterwerten aus der Konfigurationsdatei zu konfigurieren.	Die Schnittstelle ist bereit Die Konfigurationsdatei ist nicht vorhanden	Fehlermeldung wird angezeigt Anwendung beendet	ok

Tabelle 53: Test Szenarien Vorlage herunterladen

Nr.	Szenario	Beschreibung	Ausgangslage	Erwartetes Resultat	Erfolg
VH1	Normal herunterladen	Nach dem Eröffnen eines neuen Auftrages wird in der Antwort vom PMS die URL für die Vorlage zurückgegeben Mit dieser URL wird die Vorlage vom PMS Web-Server heruntergeladen	Neuer Auftrag eröffnet URL in Antwort enthalten PMS -Web-Server erreichbar	Vorlage heruntergeladen und im Speicher	ok
VH2	URL nicht vorhanden	Nach dem Eröffnen eines neuen Auftrages wird in der Antwort vom PMS die URL für die Vorlage nicht zurückgegeben	Neuer Auftrag eröffnet URL in Antwort nicht enthalten	Fehlermeldung wird angezeigt Möglichkeit bieten, mit lokaler Vorlagendatei weiterzufahren Möglichkeit bieten, Auftrag abzubrechen	ok
VH3	PMS Web-Server nicht erreichbar	Nach dem Eröffnen eines neuen Auftrages wird in der Antwort vom PMS die URL für die Vorlage zurückgegeben Mit dieser URL wird versucht, die Vorlage vom PMS Web-Server herunterzuladen	Neuer Auftrag eröffnet URL in Antwort enthalten PMS -Web-Server nicht erreichbar	Fehlermeldung wird angezeigt Möglichkeit bieten, erneuter Druckvorgang zu starten Möglichkeit bieten, Auftrag abzubrechen	ok
VH4	URL ungültig	Nach dem Eröffnen eines neuen Auftrages wird in der Antwort vom PMS eine ungültige URL für die Vorlage zurückgegeben	Neuer Auftrag eröffnet Ungültige URL in Antwort enthalten	Fehlermeldung wird angezeigt Möglichkeit bieten, mit lokaler Vorlagendatei weiterzufahren Möglichkeit bieten, Auftrag abzubrechen	ok

Tabelle 54: Test Szenarien Vorlage speichern

Nr.	Szenario	Beschreibung	Ausgangslage	Erwartetes Resultat	Erfolg
VH1	Normal speichern	Die heruntergeladene Vorlage lokal speichern	Vorlage im Speicher Schreibrecht im entsprechenden Verzeichnis	Vorlage lokal gespeichert	ok
VH2	Kein Schreibrecht	Versuch, die heruntergeladene Vorlage ohne Schreibrecht lokal abzuspeichern	Vorlage im Speicher Kein Schreibrecht im entsprechenden Verzeichnis	Fehlermeldung wird angezeigt Möglichkeit bieten, erneuter Druckvorgang zu starten Möglichkeit bieten, Auftrag abzubrechen	ok

Tabelle 55: Test Szenarien Vorlageninhalt interpretieren

Nr.	Szenario	Beschreibung	Ausgangslage	Erwartetes Resultat	Erfolg
VII1	Normal interpretieren	Die gespeicherte Vorlage einlesen, parsen und interpretieren	Auftrag offen Teilarbeit beendet Label soll ausgedruckt werden Vorlagendatei ohne inhaltliche Fehler vorhanden	Druckinhalt vorbereitet zum ausdrucken	ok
VII2	Vorlagendatei nicht vorhanden	Versuch, fehlende Vorlagendatei zu öffen	Auftrag offen Teilarbeit beendet Label soll ausgedruckt werden Vorlagendatei nicht vorhanden	Fehlermeldung wird angezeigt Möglichkeit bieten, erneuter Druckvorgang zu starten Möglichkeit bieten, Auftrag abzubrechen	ok
VII3	Kein Leserecht	Versuch, ohne Leserecht die Vorlagendatei zu öffnen	Auftrag offen Teilarbeit beendet Label soll ausgedruckt werden Vorlagendatei ohne Leserecht vorhanden	Fehlermeldung wird angezeigt Möglichkeit bieten, erneuter Druckvorgang zu starten Möglichkeit bieten, Auftrag abzubrechen	ok
VII4	Fehler in Vorlage	Die gespeicherte Vorlage einlesen, parsen und interpretieren, obschon Fehler in der Vorlage	Auftrag offen Teilarbeit beendet Label soll ausgedruckt werden Vorlagendatei mit mindestens 1 inhaltlichen Fehler vorhanden	Fehlermeldung wird angezeigt Möglichkeit bieten, erneuter Druckvorgang zu starten Möglichkeit bieten, Auftrag abzubrechen	ok

Tabelle 56: Test Szenarien Label drucken

Nr.	Szenario	Beschreibung	Ausgangslage	Erwartetes Resultat	Erfolg
LD1	Normal drucken	Die gespeicherte Vorlage einlesen, parsen und interpretieren	Druckinhalt vorhanden Hardware-Schnittstelle bereit Drucker bereit	Label ausgedruckt	fail
LD2	Hardware-Schnittstelle kann nicht geöffnet werden	Der Druckinhalt ist vorbereitet zum Ausdruck, jedoch kann nicht gedruckt werden, weil die Hardware-Schnittstelle nicht geöffnet werden kann	Druckinhalt vorhanden Hardware-Schnittstelle nicht bereit	Fehlermeldung wird angezeigt Möglichkeit bieten, erneuter Druckvorgang zu starten Möglichkeit bieten, Auftrag abzubrechen	fail
LD3	Drucker nicht bereit	Der Druckinhalt ist vorbereitet zum Ausdruck, jedoch kann nicht gedruckt werden, weil der Drucker nicht bereit ist	Druckinhalt vorhanden Hardware-Schnittstelle bereit Drucker nicht bereit	Fehlermeldung wird angezeigt Möglichkeit bieten, erneuter Druckvorgang zu starten Möglichkeit bieten, Auftrag abzubrechen	fail
LD4	Drucker im Fehlerzustand	Der Druckinhalt ist vorbereitet zum Ausdruck, jedoch kann nicht gedruckt werden, weil der Drucker im Fehlerzustand ist	Druckinhalt vorhanden Hardware-Schnittstelle bereit Drucker im Fehlerzustand	Fehlermeldung wird angezeigt Möglichkeit bieten, erneuter Druckvorgang zu starten Möglichkeit bieten, Auftrag abzubrechen	fail

12.3.1.2 Barcode-Leser

Tabelle 57: Test Szenarien Barcode-Leser konfigurieren

Nr.	Szenario	Beschreibung	Ausgangslage	Erwartetes Resultat	Erfolg
BLK1	Normal konfiguriert	Die Anwendung startet und die Hardware-Schnittstelle zum Barcode-Leser wird mittels den Parameterwerten aus der Konfigurationsdatei konfiguriert	Die Schnittstelle ist bereit Die Konfigurationsdatei ist vorhanden Die Parameterwerte in der Konfigurationsdatei sind korrekt	Hardware-Schnittstelle konfiguriert Anwendung initialisiert weiter	ok
BLK2	Schnittstelle besetzt	Die Anwendung startet und versucht die Hardware-Schnittstelle zum Barcode-Leser mittels den Parameterwerten aus der Konfigurationsdatei zu konfigurieren.	Die Schnittstelle ist nicht bereit Die Konfigurationsdatei ist vorhanden Die Parameterwerte in der Konfigurationsdatei sind korrekt	Fehlermeldung wird angezeigt Anwendung beendet	ok
BLK3	Falsche Parameterwerte in Konfigurationsdatei	Die Anwendung startet und versucht die Hardware-Schnittstelle zum Barcode-Leser mittels den Parameterwerten aus der Konfigurationsdatei zu konfigurieren.	Die Schnittstelle ist bereit Die Konfigurationsdatei ist vorhanden Mindestens 1 Parameterwert in der Konfigurationsdatei ist falsch	Fehlermeldung wird angezeigt Anwendung beendet	ok
BLK4	Konfigurationsdatei fehlt	Die Anwendung startet und versucht die Hardware-Schnittstelle zum Barcode-Leser mittels den Parameterwerten aus der Konfigurationsdatei zu konfigurieren.	Die Schnittstelle ist bereit Die Konfigurationsdatei ist nicht vorhanden	Fehlermeldung wird angezeigt Anwendung beendet	ok

Tabelle 58: Test Szenarien Barcode lesen

Nr.	Szenario	Beschreibung	Ausgangslage	Erwartetes Resultat	Erfolg
BL1	Brick-Barcode einlesen	Der Operator XY liest mit dem Barcode-Leser den Barcode eines Bricks ein	Barcode-Leser bereit	Barcode ist eingelesen Barcode weitergeleitet	ok
BL2	Hardware-Schnittstelle kann nicht geöffnet werden	Der Operator XY liest mit dem Barcode-Leser den Barcode eines Bricks ein	Barcode-Leser bereit Hardware-Schnittstelle ist blockiert	Fehlermeldung wird angezeigt Möglichkeit bieten, erneuter Lesevorgang zu starten Möglichkeit bieten, Auftrag abzubrechen Möglichkeit bieten, Barcode von Hand eingeben	ok
BL3	Leser nicht bereit	Der Operator XY liest mit dem Barcode-Leser den Barcode eines Bricks ein	Barcode-Leser nicht bereit	System wartet auf Barcode vom Leser	ok

12.3.2 Framework

12.3.2.1 Entwickler

Tabelle 59: Test Szenarien HW-Adapter hinzufügen

Nr.	Szenario	Beschreibung	Ausgangslage	Erwartetes Resultat	Erfolg
HWAH1	HW-Adapter normal hinzufügen.	Der Entwickler trägt in der Konfigurationsdatei der Terminalanwendung den HW-Adapter inkl. dem Pfad für die Konfigurationsdatei des HW-Adapters ein. Die Bibliothek wird in denselben Ordner gespeichert, wie die Anwendung.	Die Bibliothek für den HW-Adapter und die dazugehörige Konfigurationsdatei ist am richtigen Ort gespeichert. Der Namen der Bibliotheke stimmt. Der Pfad zur dazugehörigen Konfigurationsdatei stimmt.	Die Bibliothek und die Konfigurationsdatei ist am richtigen Ort abgespeichert.	ok

Tabelle 60: Test Szenarien Menü bearbeiten

Nr.	Szenario	Beschreibung	Ausgangslage	Erwartetes Resultat	Erfolg
MEB1	Menüpunkt erstellen	Es können beliebig viele neue Menüpunkte erstellt werden.	Es besteht eine neue Seite	Zwei neue Menüpunkte sind eingefügt.	ok
MEB2	Menüpunkt bearbeiten	Der Entwickler tauscht zwei Menüpunkt aus.	Mindestens zwei Menüpunkte bestehen	Die Menüpunkte sind vertauscht.	ok
MEB3	Menüpunkt löschen	Der Entwickler löscht einen erstellten Menüpunkt	Es besteht mindestens ein Menüpunkt.	Der Menüpunkt ist gelöscht.	ok

Tabelle 61: Test Szenarien Fensterinhalt erstellen

Nr.	Szenario	Beschreibung	Ausgangslage	Erwartetes Resultat	Erfolg
FIE1	Fensterinhalt (Text) erstellen	Der Entwickler fügt einen neuen Text in einer neuen Seite ein.	Eine neue Seite ist hinzugefügt worden	In der neuen Seite ist der eingegebene Text vorhanden	ok
FIE2	Fensterinhalt (Bild) erstellen	Der Entwickler fügt ein neues Bild in der neuen Seite ein	Eine neue Seite ist hinzugefügt worden	In der neuen Seite ist das ausgewählte Bild vorhanden	ok
FIE3	Fensterinhalt (Text) bearbeiten	Der Entwickler ändert den vorhanden Text	Eine bestehenden Seite mit Text ist vorhanden	In der gegebenen Seite ist der geänderte Text vorhanden	ok
FIE4	Fensterinhalt (Bild) bearbeiten	Der Entwickler ersetzt ein bestehendes Bild	Eine bestehende Seite mit einem Bild ist vorhanden	In der gegebenen Seite ist das neue Bild vorhanden	ok
FIE5	Fensterinhalt ist zu gross	Der Entwickler erstellt in einer Seite viel Inhalt mit Bildern und Texten, so dass der Inhalt die Seitengrösse überschreitet.	Es besteht eine Seite	Es wird automatisch eine Scroll-Bar erzeugt	ok

Tabelle 62: Test Szenarien Fenster hinzufügen

Nr.	Szenario	Beschreibung	Ausgangslage	Erwartetes Resultat	Erfolg
FH1	Fenster hinzufügen	Der Entwickler fügt eine neue Seite ein	Das Grundgerüst für eine Terminalanwendung ist vorhanden	Eine neue Seite ist eingefügt	ok

Tabelle 63: Test Szenarien Sprache hinzufügen

Nr.	Szenario	Beschreibung	Ausgangslage	Erwartetes Resultat	Erfolg
SH1	Sprachen hinzufügen	Der Entwickler fügt eine neue Sprache hinzu.	Es besteht eine Sprachtabelle für die neue Sprache.	Die Sprache ist hinzugefügt	ok

Tabelle 64: Test Szenarien Bilder hinzufügen

Nr.	Szenario	Beschreibung	Ausgangslage	Erwartetes Resultat	Erfolg
BH1	Bilder hinzufügen	Der Entwickler fügt neue Bilder ein	Es besteht ein Fenster, worin die eingefügten Bilder angezeigt werden können.	Die eingefügten Bilder werden in der vorgegebenen Reihenfolge und richtigen Grösse angezeigt	ok
BH2	Grössere Bilder hinzufügen	Der Entwickler fügt neue Bilder ein, welche zu gross sind.	Einzelne oder alle Bilder sind zu gross.	Die eingefügten Bilder werden zu gross angezeigt	ok
BH3	Kleinere Bilder hinzufügen	Der Entwickler fügt neue Bilder ein, welche zu klein sind.	Einzelne oder alle Bilder sind zu klein.	Die eingefügten Bilder werden zu klein angezeigt	ok

Tabelle 65: Test Szenarien Barcode de-aktivieren

Nr.	Szenario	Beschreibung	Ausgangslage	Erwartetes Resultat	Erfolg
BDA1	Barcodetyp aktivieren	Der Entwickler aktiviert einen Barcodetyp in der Konfigurationsdatei.	Der Barcodetyp ist in der Konfigurationsdatei vorhanden	Der neue Barcodetyp ist aktiviert	ok
BDA2	Barcodetyp deaktivieren	Der Entwickler deaktiviert einen Barcodetyp in der Konfigurationsdatei.	Der Barcodetyp ist in der Konfigurationsdatei vorhanden	Der neue Barcodetyp ist nicht aktiviert	ok

Tabelle 66: Test Szenarien Fenster de-aktivieren

Nr.	Szenario	Beschreibung	Ausgangslage	Erwartetes Resultat	Erfolg
FDA1	Fenster aktivieren	Der Entwickler aktiviert das Fenster in der Konfigurationsdatei	Das Fenster ist in der Konfigurationsdatei vorhanden	Das Fenster ist aktiviert	ok
FDA2	Fenster deaktivieren	Der Entwickler deaktiviert das Fenster in der Konfigurationsdatei	Das Fenster ist in der Konfigurationsdatei vorhanden	Das Fenster ist deaktiviert	ok

Tabelle 67: Test Szenarien Hilfsmittel de-aktivieren

Nr.	Szenario	Beschreibung	Ausgangslage	Erwartetes Resultat	Erfolg
HMDA1	Hilfsmittel aktivieren	Der Entwickler aktiviert das Hilfsmittel in der Konfigurationsdatei	Das Hilfsmittel ist in der Konfigurationsdatei vorhanden	Das Hilfsmittel ist aktiviert	ok
HMDA2	Hilfsmittel deaktivieren	Der Entwickler deaktiviert das Hilfsmittel in der Konfigurationsdatei	Das Hilfsmittel ist in der Konfigurationsdatei vorhanden	Das Hilfsmittel ist deaktiviert	ok

12.3.2.2 System

Tabelle 68: Test Szenarien Url speichern

Nr.	Szenario	Beschreibung	Ausgangslage	Erwartetes Resultat	Erfolg
US1	URL speichern	Die URL wird abgespeichert	Eine URL ist vorhanden	Die URL ist abgespeichert	ok

Tabelle 69: TestSzenarien Dokumente herunterladen

Nr.	Szenario	Beschreibung	Ausgangslage	Erwartetes Resultat	Erfolg
DH1	Text-/XML-/HTML-Dokument herunterladen	Ein Text-/XML-/HTML-Dokument via einer gespeicherten URL vom PMS herunterladen	Die URL ist vorhanden WebServer vom PMS ist erreichbar	Das Text-/XML-/HTML-File ist im Arbeitspeicher vorhanden	ok
DH2	URL nicht vorhanden	Ein Dokument via einer gespeicherten URL vom PMS herunterladen. Es existiert keine URL zum Herunterladen.	Die URL ist nicht vorhanden. WebServer vom PMS ist erreichbar	Fehlermeldung, dass keine URL vorhanden ist.	ok
DH3	PMS Web Server nicht erreichbar	Ein Dokument via einer gespeicherten URL vom PMS herunterladen. Der Web Server ist nicht erreichbar.	Die URL ist vorhanden. WebServer vom PMS ist nicht erreichbar	Fehlermeldung, dass der Web Server nicht erreichbar ist.	ok

Tabelle 70: Test Szenarien Dokumente anzeigen

Nr.	Szenario	Beschreibung	Ausgangslage	Erwartetes Resultat	Erfolg
DA1	Text-/XML-/HTML-Dokument anzeigen	Das im Arbeitsspeicher vorhandene Dokument wird im dafür vorgesehenen Fenster angezeigt.	Das Dokument ist im Arbeitsspeicher gespeichert.	Das Dokument wird im gewünschten Fenster angezeigt	ok
DA2	Text-/XML-/HTML-Dokument nicht vorhanden	Im Arbeitsspeicher ist kein Dokument vorhanden	Das Dokument ist nicht im Arbeitsspeicher gespeichert.	Es erscheint eine Fehlermeldung, dass keine Dokumente vorhanden sind	ok

Tabelle 71: Test Szenarien Auftrags-ID senden

Nr.	Szenario	Beschreibung	Ausgangslage	Erwartetes Resultat	Erfolg
AIS1	Auftrags-ID senden	Eine gültige Auftrags-ID ans PMS senden.	Eine Auftrags-ID ist vorhanden. AS läuft.	Die Auftrags-ID ist beim PMS angekommen. Das PMS sendet eine Antwort mit der Bestätigung, dass die Antwort korrekt ist.	ok
AIS2	Ungültige Auftrags-ID ans PMS senden	Eine ungültige Auftrags-ID ans PMS senden.	Eine Auftrags-ID ist vorhanden. AS läuft.	Die Auftrags-ID ist beim PMS angekommen. Das PMS sendet eine Antwort mit einer Meldung, dass die ID nicht korrekt ist.	ok
AID3	AS kann nicht erreicht werden	Eine gültige Auftrags-ID ans PMS senden. Der AS ist nicht erreichbar.	Eine Auftrags-ID ist vorhanden. AS läuft nicht.	Es erscheint eine Fehlermeldung, dass der AS nicht erreichbar ist	ok

Tabelle 72: Test Szenarien ID-Antwort speichern

Nr.	Szenario	Beschreibung	Ausgangslage	Erwartetes Resultat	Erfolg
IAS1	ID-Antwort speichern	Nach dem Senden der Auftrags-ID, kommen mit der Antwort die benötigten Auftragsdaten zurück. Diese Daten werden gespeichert.	Die Auftrags-ID ist dem PMS zugestellt worden	Die benötigten Daten sind im Arbeitsspeicher abgespeichert.	ok
IAS2	ID-Antwort nicht vorhanden	Nach dem Senden der Auftrags-ID, kommen mit der Antwort keine Auftragsdaten zurück.	Die Auftrags-ID ist dem PMS zugestellt worden	Es erscheint eine Fehlermeldung, dass keine Auftragsdaten vorhanden sind.	ok
IAS3	Keine Antwort	Nach dem Senden der Auftrags-ID, kommt keine Antwort.	Die Auftrags-ID ist dem PMS zugestellt worden	Es erscheint eine Fehlermeldung, dass keine Antwort vom PMS zurückkommt	ok

Tabelle 73: Test Szenarien Barcode einlesen

Nr.	Szenario	Beschreibung	Ausgangslage	Erwartetes Resultat	Erfolg
BE1	Barcode einlesen	Das System erhält vom HW-Adapter eine Meldung, dass ein Barcode vorhanden ist. Das System liest den Barcode ein und speichert ihn im Arbeitsspeicher ab.	Es kommt eine Meldung vom HW-Adapter, dass ein Barcode vorhanden ist.	Der Wert des eingelesenen Barcodes ist im Arbeitspeicher gespeichert.	ok
BE2	Ungültiger Barcode einlesen	Das System erhält vom HW-Adapter eine Meldung, dass ein Barcode vorhanden ist. Das System liest den Barcode ein und merkt, dass der Barcode nicht gültig ist.	Es kommt eine Meldung vom HW-Adapter, dass ein Barcode vorhanden ist.	Es erscheint eine Fehlermeldung, dass der Barcode ungültig ist.	ok
BE3	Kein Barcode vorhanden	Das System erhält vom HW-Adapter eine Meldung, dass ein Barcode vorhanden ist. Nach dem Lesen ist kein Barcode vorhanden.	Es kommt eine Meldung vom HW-Adapter, dass ein Barcode vorhanden ist.	Es erscheint eine Fehlermeldung, dass kein Barcode vorhanden ist.	ok

Tabelle 74: Test Szenarien Barcode aufsplitten

Nr.	Szenario	Beschreibung	Ausgangslage	Erwartetes Resultat	Erfolg
BA1	Barcode aufsplitten	Nach dem Einlesen wird der Barcode nach Typ identifiziert und ins fokusierte Feld eingetragen. Nach dem Drücken der Enter-Taste wird der Wert übernommen.	Es ist ein Barcode im Arbeitsspeicher vorhanden.	Nach dem Einlesen des Barcodes und dem Bestätigen mit Enter wird der Barcode übernommen.	ok
BA2	Unbekannter Barcode-Typ	Nach dem Einlesen wird der Barcode nach Typ identifiziert und ins fokusierte Feld eingetragen. Nach dem Drücken der Enter-Taste wird der Wert nicht übernommen.	Es ist ein Barcode im Arbeitsspeicher vorhanden.	Nach dem Einlesen des Barcodes und dem Bestätigen mit Enter erscheint eine Fehlermeldung, dass der Barcode-Typ unbekannt ist.	ok
BA3	Ungültiger Barcode-Typ	Nach dem Einlesen wird der Barcode nach Typ identifiziert und ins fokusierte Feld eingetragen. Nach dem Drücken der Enter-Taste wird der Wert nicht übernommen.	Es ist ein Barcode im Arbeitsspeicher vorhanden.	Nach dem Einlesen des Barcodes und dem Bestätigen mit Enter erscheint eine Fehlermeldung, dass der Barcode-Typ ungültig ist (bekannt, aber z.B. nicht aktiv).	ok

Tabelle 75: Test Szenarien Status speichern

Nr.	Szenario	Beschreibung	Ausgangslage	Erwartetes Resultat	Erfolg
SS1	Status speichern	Der Status des Produktionsstands wird gespeichert und ans PMS gesendet	Der AS ist bereit	Der neue Status ist im PMS und in der Terminalanwendung gespeichert.	ok
SS2	Status speichern, AS nicht erreichbar	Der Status des Produktionsstands wird gespeichert und ans PMS gesendet. Der AS ist jedoch nicht erreichbar.	Der AS ist nicht bereit	Der neue Status ist bei der Terminalanwendung gespeichert. Beim PMS jedoch nicht.	ok

Tabelle 76: Test Szenarien HW-Adapter laden

Nr.	Szenario	Beschreibung	Ausgangslage	Erwartetes Resultat	Erfolg
HAL1	HW-Adapter laden	Beim Starten der Terminalanwendung werden die aktivierten HW-Adapter geladen und konfiguriert.	Die Bibliothek ist vorhanden. Die dazugehörige Konfigurationsdatei ist vorhanden. Der konfigurierte Pfad stimmt.	Der HW-Adapter ist eingebunden und konfiguriert.	ok
HAL2	HW-Adapter laden, jedoch ohne Konfigurationsdatei.	Beim Starten der Terminalanwendung wird ein HW-Adapter geladen. Er kann nicht eingebunden werden, da die Konfigurationsdatei fehlt.	Die Bibliothek ist vorhanden. Die dazugehörige Konfigurationsdatei ist nicht vorhanden. Der konfigurierte Pfad stimmt.	Es erscheint eine Fehlermeldung, dass die Konfigurationsdatei fehlt.	ok
HAL3	HW-Adapter laden, mit falschem Konig-Pfad.	Beim Starten der Terminalanwendung wird ein HW-Adapter geladen. Der Pfad für die Konfigurationsdatei des HW-Adapters ist falsch.	Die Bibliothek ist vorhanden. Die dazugehörige Konfigurationsdatei ist vorhanden. Der konfigurierte Pfad ist falsch.	Es erscheint eine Fehlermeldung, dass die Konfigurationsdatei fehlt.	ok
HAL4	HW-Adapter laden, Bibiliotheke nicht vorhanden	Beim Starten der Terminalanwendung wird versucht einen HW-Adapter einzubinden.	Es ist keine Bibliothek vorhanden. Es ist eine Konfigurationsdatei vorhanden. Der konfigurierte Pfad stimmt.	Es wird kein HW-Adapter eingebunden und es erscheint eine Fehlermeldung.	ok
HAL5	HW-Adapter laden, Falscher Bibiliothek-Namen	Beim Starten der Terminalanwendung wird versucht einen HW-Adapter einzubinden.	Es ist eine Bibliothek vorhanden. Es ist eine Konfigurationsdatei vorhanden. Der konfigurierte Pfad stimmt. Der konfigurierte Name des HW-Adapters ist falsch.	Es wird kein HW-Adapter eingebunden und es erscheint eine Fehlermeldung.	ok

12.3.2.3 Operator

Tabelle 77: Test Szenarien Sprache ändern

Nr.	Szenario	Beschreibung	Ausgangslage	Erwartetes Resultat	Erfolg
SAE1	Sprache ändern	Der Operator wechselt die Sprache.	Die Sprache wurde hinzugefügt	Die komplette Anwendung kann online von Deutsch auf Englisch und unbekehrt umgestellt werden	ok
SAE2	Nicht vollständige Sprachtabelle	Der Operator ändert eine Sprache mit einer unvollständigen Sprachtabelle.	Es besteht eine unvollständige Sprachtabelle Die Sprache wurde hinzugefügt	Die fehlenden Texte in der Sprachtabelle werden in der Standard-Sprache EN dargestellt	ok
SAE3	Nicht lateinische Sprache hinzufügen	Der Operator ändert die Sprache auf Chinesisch.	Es besteht eine Sprachtabelle für Chinesisch Die Sprache wurde hinzugefügt	Der Zeichensatz für China wird richtig dargestellt	ok

Tabelle 78: Test Szenarien Sprachabhängige Bilder anzeigen

Nr.	Szenario	Beschreibung	Ausgangslage	Erwartetes Resultat	Erfolg
SABA1	Sprachabhängige Bilder anzeigen	Die sprachabhängigen Bilder werden anhand der gewählten Sprache anzeigen	Die Bilder für die gewählte Sprache sind vorhanden	Die Bilder, welche sprachabhängig sind werden korrekt angezeigt	ok
SABA2	Sprachabhängige Bilder nicht vorhanden	Die sprachabhängigen Bilder können anhand der gewählten Sprache nicht angezeigt werden	Die Bilder für die gewählte Sprache sind vorhanden	Die Bilder, welche sprachabhängig sind werden werden in der Standardsprache EN angezeigt. Sind auch da keine Bilder vorhanden, erscheint eine Fehlermeldung.	ok

Tabelle 79: Test Szenarien Operator-ID eingeben

Nr.	Szenario	Beschreibung	Ausgangslage	Erwartetes Resultat	Erfolg
OI1	Operator-ID von Hand eingeben	Nach dem Öffnen des Anmeldefensters wird das Eingabefeld fokusiert. Der Operator gibt via Tastatur oder virtueller Tastatur seine ID ein und drückt Enter.	Es wird eine Anmeldung vom System verlangt oder der Operator will seine ID eingeben (Bsp. Operator-Wechsel).	Die Anwendung hat die Identifikationsnummer des Operators übernommen. Die virtuelle Tastatur schliesst.	ok
OI2	Operator-ID mittels Barcode-Leser eingeben	Nach dem Öffnen des Anmeldefensters wird das Eingabefeld fokusiert. Der Operator gibt mittels eines Barcode-Lesers seine ID ein.	Es wird eine Anmeldung vom System verlangt oder der Operator will seine ID eingeben (Bsp. Operator-Wechsel). Der Barcode-Leser ist bereit.	Die Bestätigung geschieht automatisch und der Identifikationsnummer des Operators ist übernommen worden.	ok
OI3	Falsche ID	Nach dem Öffnen des Anmeldefensters wird das Eingabefeld fokusiert. Der Operator gibt eine falsche ID ein und drückt Enter.	Es wird eine Anmeldung vom System verlangt oder der Operator will seine ID eingeben (Bsp. Operator-Wechsel).	Die Identifikationsnummer des Operators wird nicht übernommen. Es erscheint eine Fehlermeldung.	ok

Tabelle 80: Test Szenarien Operator-ID abbrechen

Nr.	Szenario	Beschreibung	Ausgangslage	Erwartetes Resultat	Erfolg
OIA1	Eingabe der Operator-ID wird abgebrochen	Nach dem Öffnen des Anmeldefensters wird das Eingabefeld fokusiert. Der Operator gibt eine ID ein und drückt Abbrechen.	Es wird eine Anmeldung vom System verlangt oder der Operator will seine ID eingeben (Bsp. Operator-Wechsel).	Die Identifikationsnummer des Operators wird nicht übernommen.	ok

Tabelle 81: Test Szenarien Fenster öffnen

Nr.	Szenario	Beschreibung	Ausgangslage	Erwartetes Resultat	Erfolg
FOE1	Arbeitsbeschreibungshilfe öffnen	Der Operator öffnet das Fenster.	Das Fenster ist aktiviert und in einem Menüpunkt hinterlegt. Das Dokument/Beschreibung ist hinterlegt.	Das Fenster wird angezeigt und ist modal offen.	ok
FOE2	Arbeitsbeschreibungshilfe ohne hinterlegte Dokumente öffnen	Der Operator öffnet das Fenster.	Das Fenster ist aktiviert und in einem Menüpunkt hinterlegt. Das Dokument/Beschreibung ist nicht hinterlegt.	Es erscheint eine Fehlermeldung	ok
FOE3	Auftragsinformationen öffnen	Der Operator öffnet das Fenster.	Das Fenster ist aktiviert und in einem Menüpunkt hinterlegt. Das Dokument/Beschreibung ist hinterlegt.	Das Fenster wird angezeigt und ist modal offen.	ok
FOE4	Auftragsinformationen ohne hinterlegte Dokumente öffnen	Der Operator öffnet das Fenster.	Das Fenster ist aktiviert und in einem Menüpunkt hinterlegt. Das Dokument/Beschreibung ist nicht hinterlegt.	Es erscheint eine Fehlermeldung	ok
FOE5	Informationen zum nächsten Auftrag öffnen	Der Operator öffnet das Fenster.	Das Fenster ist aktiviert und in einem Menüpunkt hinterlegt. Das Dokument/Beschreibung ist hinterlegt.	Das Fenster wird angezeigt und ist modal offen.	ok
FOE6	Informationen zum nächsten Auftrag ohne hinterlegte Dokumente öffnen	Der Operator öffnet das Fenster.	Das Fenster ist aktiviert und in einem Menüpunkt hinterlegt. Das Dokument/Beschreibung ist nicht hinterlegt.	Es erscheint eine Fehlermeldung	ok
FOE7	Wartungsinformationen öffnen	Der Operator öffnet das Fenster.	Das Fenster ist aktiviert und in einem Menüpunkt hinterlegt. Das Dokument/Beschreibung ist hinterlegt.	Das Fenster wird angezeigt und ist modal offen.	ok
FOE8	Wartungsinformationen ohne hinterlegte Dokumente öffnen	Der Operator öffnet das Fenster.	Das Fenster ist aktiviert und in einem Menüpunkt hinterlegt. Das Dokument/Beschreibung ist nicht hinterlegt.	Es erscheint eine Fehlermeldung	ok

FOE9	Bildschirm-Reinigerhilfe öffnen	Der Operator öffnet das Fenster.	Das Fenster ist aktiviert und in einem Menüpunkt hinterlegt. Das Dokument/Beschreibung ist hinterlegt.	Das Fenster wird angezeigt und ist modal offen.	Ok
FOE10	Bildschirm-Reinigerhilfe ohne hinterlegte Dokumente öffnen	Der Operator öffnet das Fenster.	Das Fenster ist aktiviert und in einem Menüpunkt hinterlegt. Das Dokument/Beschreibung ist nicht hinterlegt.	Es erscheint eine Fehlermeldung	Ok
FOE11	Bilder-Assistent öffnen	Der Operator öffnet das Fenster.	Das Fenster ist aktiviert und in einem Menüpunkt hinterlegt. Das Dokument/Beschreibung ist hinterlegt.	Das Fenster wird angezeigt und ist modal offen.	Ok
FOE12	Bilder-Assistent ohne hinterlegte Dokumente öffnen	Der Operator öffnet das Fenster.	Das Fenster ist aktiviert und in einem Menüpunkt hinterlegt. Das Dokument/Beschreibung ist nicht hinterlegt.	Es erscheint eine Fehlermeldung	Ok
FOE13	Test-Bedingungshilfe öffnen	Der Operator öffnet das Fenster.	Das Fenster ist aktiviert und in einem Menüpunkt hinterlegt. Das Dokument/Beschreibung ist hinterlegt.	Das Fenster wird angezeigt und ist modal offen.	Ok
FOE14	Test-Bedingungshilfe ohne hinterlegte Dokumente öffnen	Der Operator öffnet das Fenster.	Das Fenster ist aktiviert und in einem Menüpunkt hinterlegt. Das Dokument/Beschreibung ist nicht hinterlegt.	Es erscheint eine Fehlermeldung	Ok
FOE15	Rechner öffnen	Der Operator öffnet das Fenster.	Das Fenster ist aktiviert und in einem Menüpunkt hinterlegt. Das Dokument/Beschreibung ist hinterlegt.	Das Fenster wird angezeigt und ist modal offen.	Ok
FOE16	Kalender öffnen	Der Operator öffnet das Fenster.	Das Fenster ist aktiviert und in einem Menüpunkt hinterlegt. Das Dokument/Beschreibung ist hinterlegt.	Das Fenster wird angezeigt und ist modal offen.	Ok
FOE17	Software-Version öffnen	Der Operator öffnet das Fenster.	Das Fenster ist aktiviert und in einem Menüpunkt hinterlegt. Das Dokument/Beschreibung ist hinterlegt.	Das Fenster wird angezeigt und ist modal offen.	Ok
FOE18	Software-Version ohne hinterlegte Dokumente öffnen	Der Operator öffnet das Fenster.	Das Fenster ist aktiviert und in einem Menüpunkt hinterlegt. Das Dokument/Beschreibung ist nicht hinterlegt.	Es erscheint eine Fehlermeldung	Ok

Tabelle 82: Test Szenarien Fenster schliessen

Nr.	Szenario	Beschreibung	Ausgangslage	Erwartetes Resultat	Erfolg
FS1	Arbeitsbeschreibungshilfe schliessen	Der Operator beendet das Fenster.	Das Fenster wird angezeigt	Das Fenster ist geschlossen	ok
FS2	Auftragsinformationen schliessen	Der Operator beendet das Fenster.	Das Fenster wird angezeigt	Das Fenster ist geschlossen	ok
FS3	Informationen zum nächsten Auftrag schliessen	Der Operator beendet das Fenster.	Das Fenster wird angezeigt	Das Fenster ist geschlossen	ok
FS4	Wartungsinformationen schliessen	Der Operator beendet das Fenster.	Das Fenster wird angezeigt	Das Fenster ist geschlossen	ok
FS5	Bildschirm-Reinigerhilfe schliessen	Der Operator beendet das Fenster.	Das Fenster wird angezeigt	Das Fenster ist geschlossen	ok
FS6	Bilder-Assistent schliessen	Der Operator beendet das Fenster.	Das Fenster wird angezeigt	Das Fenster ist geschlossen	ok
FS7	Test-Bedingungshilfe schliessen	Der Operator beendet das Fenster.	Das Fenster wird angezeigt	Das Fenster ist geschlossen	ok
FS8	Rechner schliessen	Der Operator beendet das Fenster.	Das Fenster wird angezeigt	Das Fenster ist geschlossen	ok
FS9	Kalender schliessen	Der Operator beendet das Fenster.	Das Fenster wird angezeigt	Das Fenster ist geschlossen	ok
FS10	Software-Version schliessen	Der Operator beendet das Fenster.	Das Fenster wird angezeigt	Das Fenster ist geschlossen	ok

Tabelle 83: Test Szenarien Berechnung eingeben

Nr.	Szenario	Beschreibung	Ausgangslage	Erwartetes Resultat	Erfolg
BE1	Berechnung eingeben	Der Operator gibt eine Zahl ein, anschliessend einen Operanden und wieder eine Zahl. Danach wird das Gleichheitszeichen gedrückt.	Der Rechner ist gestartet. Das Fenster ist offen.	Das richtige Resultat der Berechnung wird angezeigt.	ok

Tabelle 84: Test Szenarien SW-Update installieren

Nr.	Szenario	Beschreibung	Ausgangslage	Erwartetes Resultat	Erfolg
SUI1	Software update installieren	Der Operator führt ein Update der Software aus. Die Software wird heruntergeladen und installiert	Es ist ein Update vorhanden	Das Update ist installiert	ok
SUI2	Software update ohne vorhandenes update	Der Operator führt ein Update der Software aus. Die Software wird heruntergeladen und installiert	Es ist kein Update vorhanden	Das Update wird abgebrochen und es erscheint eine Meldung, dass kein update zur Verfügung steht	ok
SUI3	Software update installieren, Installationsabbruch	Der Operator führt ein Update der Software aus. Die Software wird heruntergeladen und kann nicht installiert werden	Es ist ein Update vorhanden	Es erscheint eine Fehlermeldung	ok

Tabelle 85: Test Szenarien Konfig-Update installieren

Nr.	Szenario	Beschreibung	Ausgangslage	Erwartetes Resultat	Erfolg
KUI1	Konfigurationsdatei update installieren	Der Operator führt ein Update der Konfigurationsdatei aus. Die Konfigurationsdatei wird heruntergeladen und installiert	Es ist ein Update vorhanden	Das Update ist installiert	ok
KUI2	Konfigurationsdatei update ohne vorhandenes update	Der Operator führt ein Update der Konfigurationsdatei aus. Die Konfigurationsdatei wird heruntergeladen und installiert	Es ist kein Update vorhanden	Das Update wird abgebrochen und es erscheint eine Meldung, dass kein Update zur Verfügung steht	ok
KUI3	Konfigurationsdatei update installieren	Der Operator führt ein Update der Konfigurationsdatei aus. Die Konfigurationsdatei wird heruntergeladen und kann nicht installiert werden	Es ist ein Update vorhanden	Es erscheint eine Fehlermeldung	ok

Tabelle 86: Test Szenarien Produktionsstand ändern

Nr.	Szenario	Beschreibung	Ausgangslage	Erwartetes Resultat	Erfolg
PAE1	Produktionsstand via Operator ändern	Der Operator übersteuert den Produktionsstand.	Die Terminalanwendung läuft.	Der Produktionsstand ist geändert	ok
PAE2	Produktionsstand via System ändern	Das System ändert den Produktionsstand	Die Terminalanwendung läuft.	Der Produktionsstand ist geändert	ok

12.4 Klassendiagramme

12.4.1 Hardware-Adapter

12.4.1.1 Label Drucker

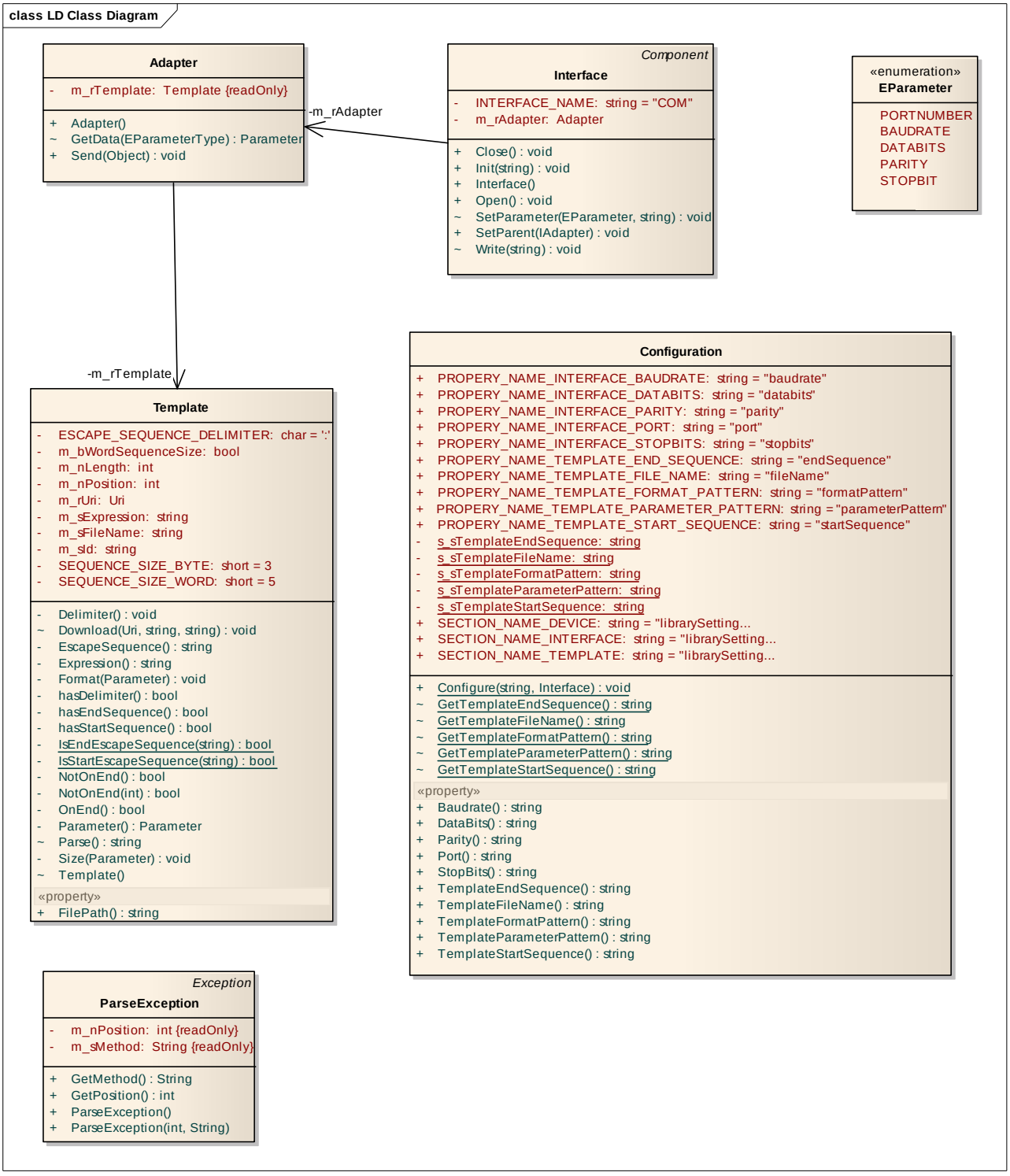


Abbildung 19: Klassendiagramm Label Drucker

12.4.1.2 Barcode-Leser

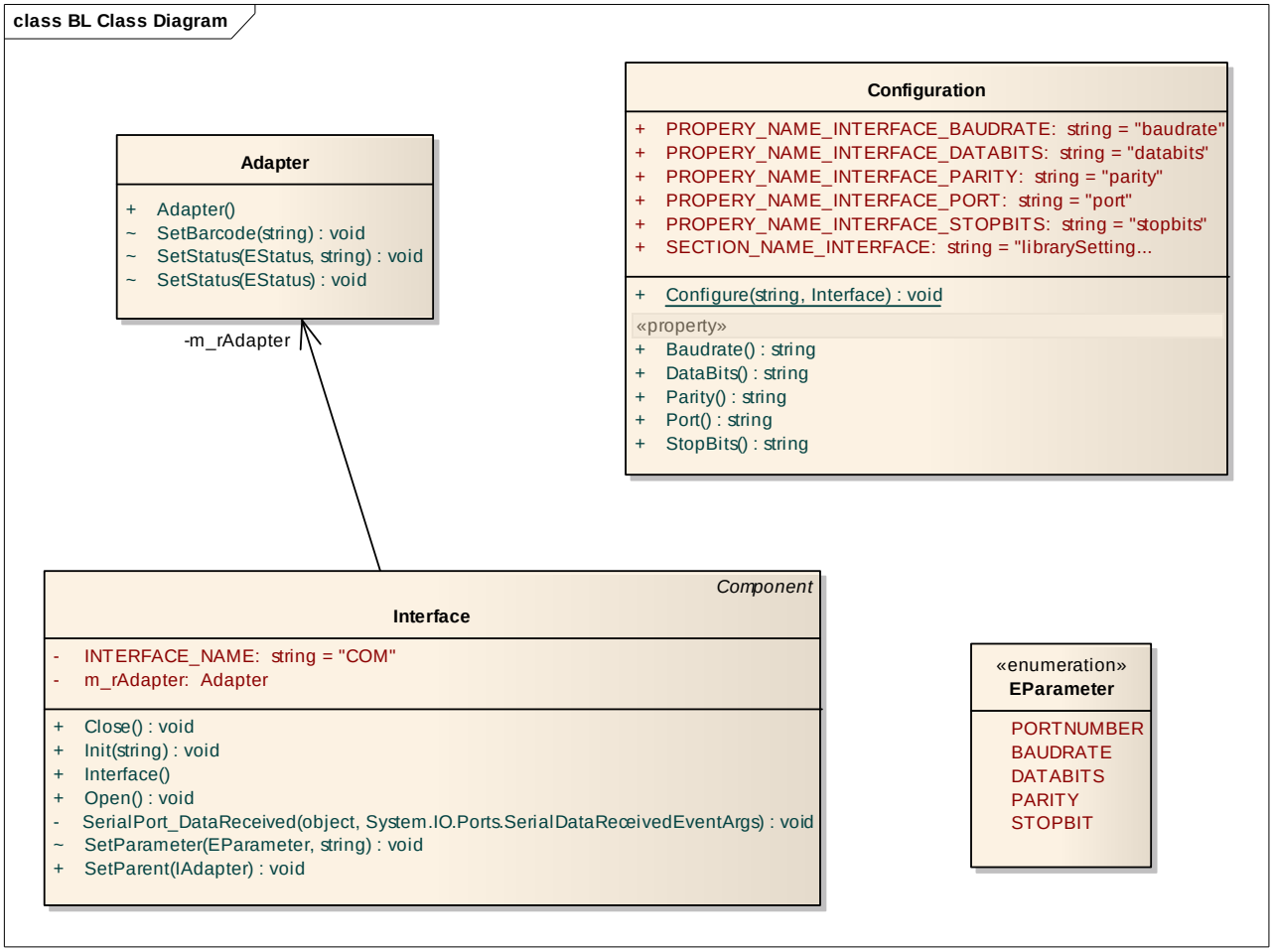


Abbildung 20: Klassendiagramm Barcode-Leser

12.4.2 Framework

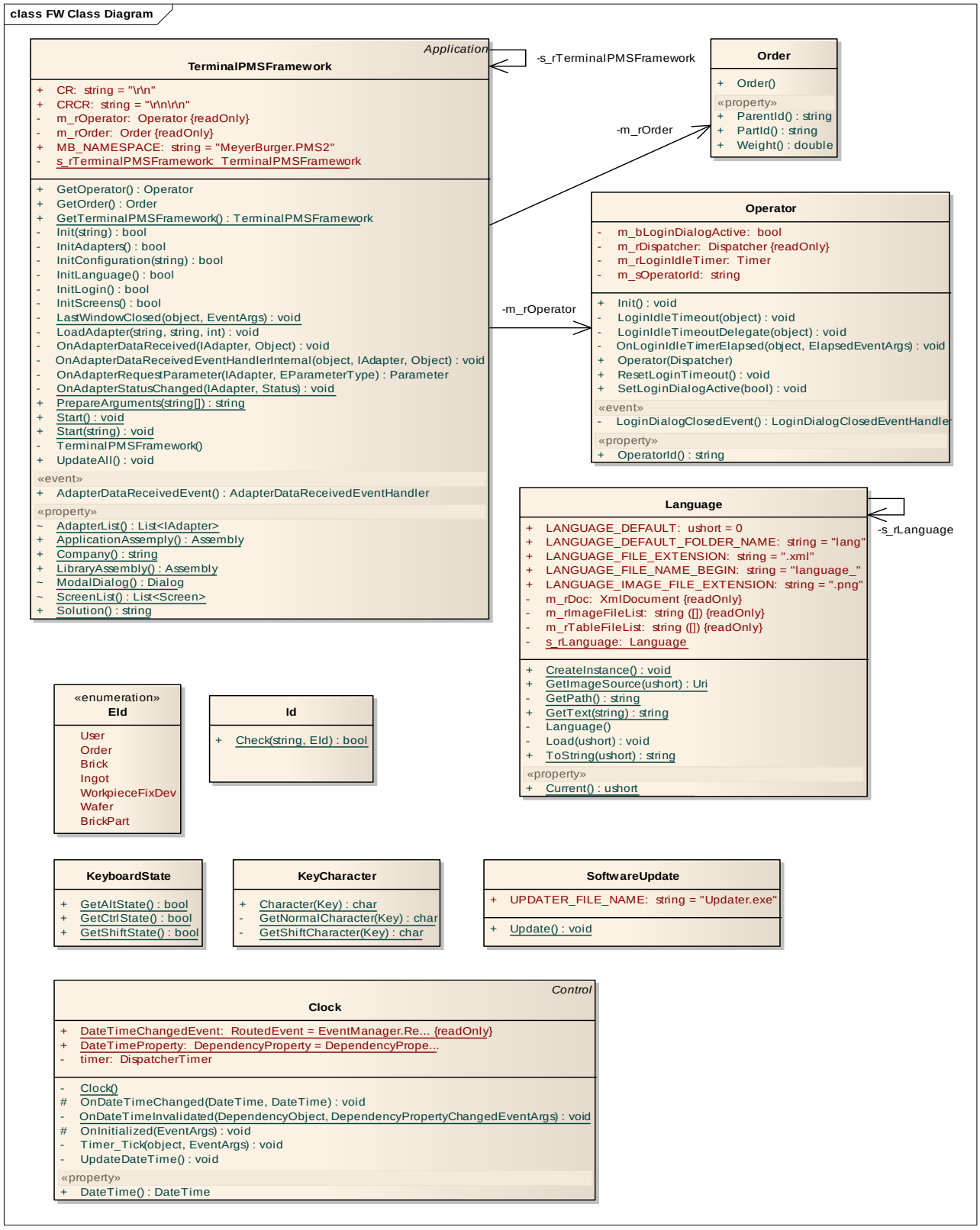


Abbildung 21: Klassendiagramm Framework, Teil 1

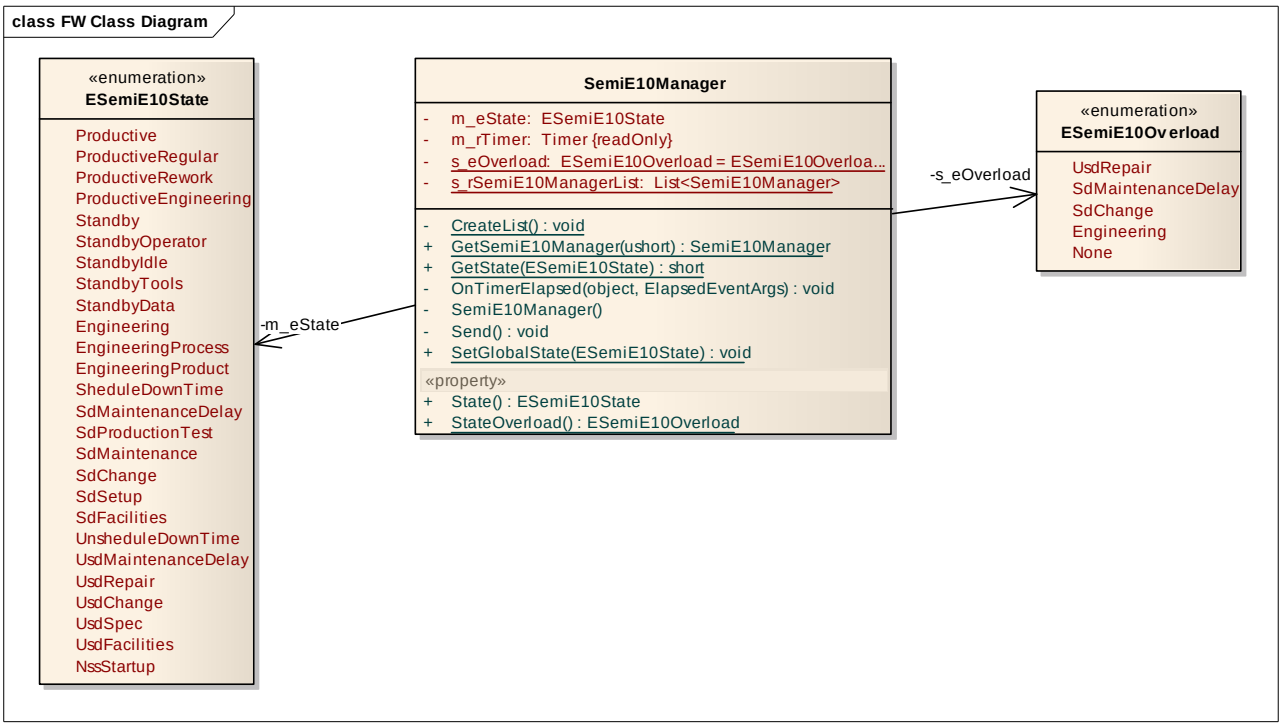


Abbildung 22: Klassendiagramm Framework, Teil 2

12.4.2.1 Adapter

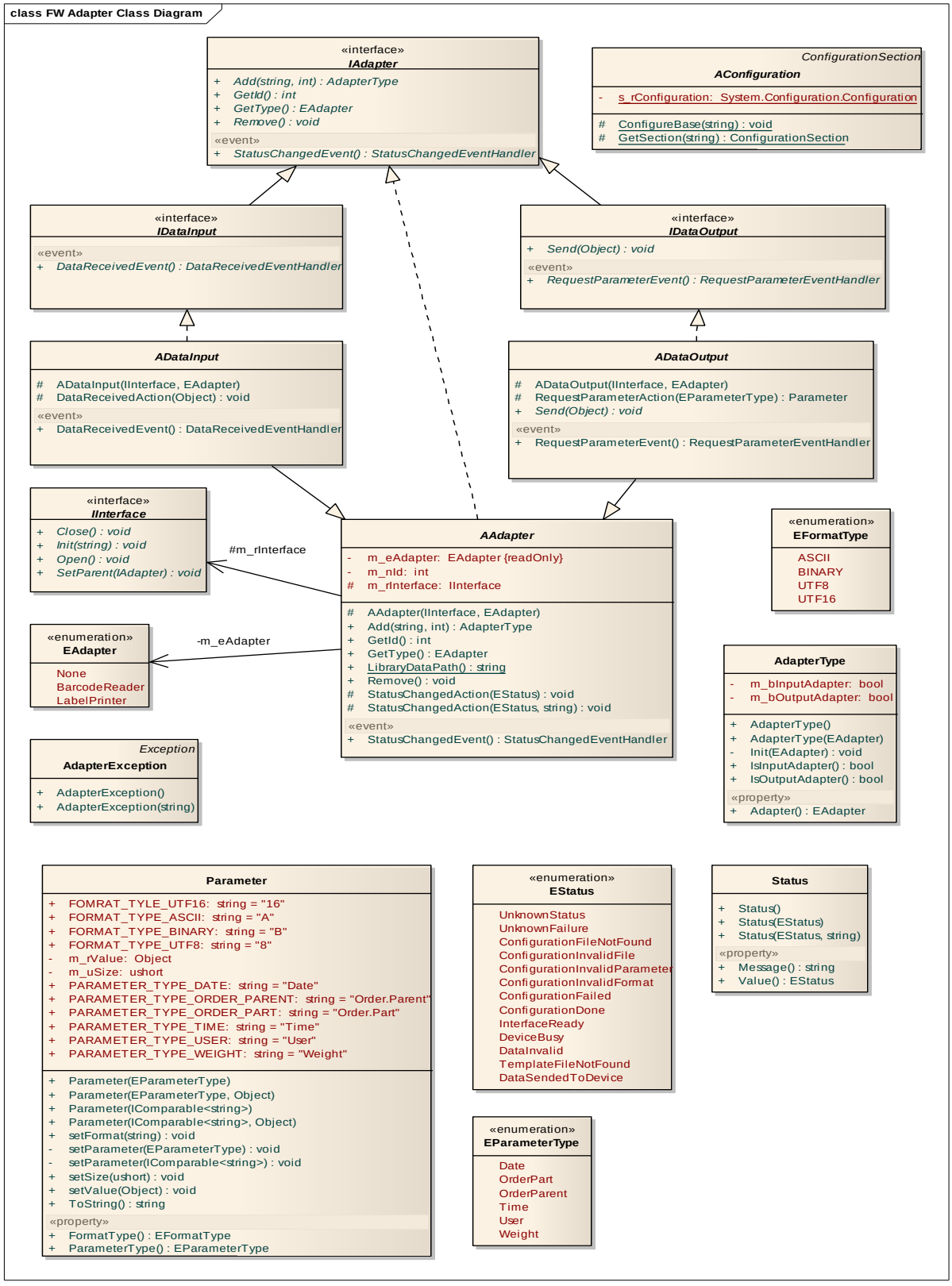


Abbildung 23: Klassendiagramm Adapter

12.4.2.2 Communication

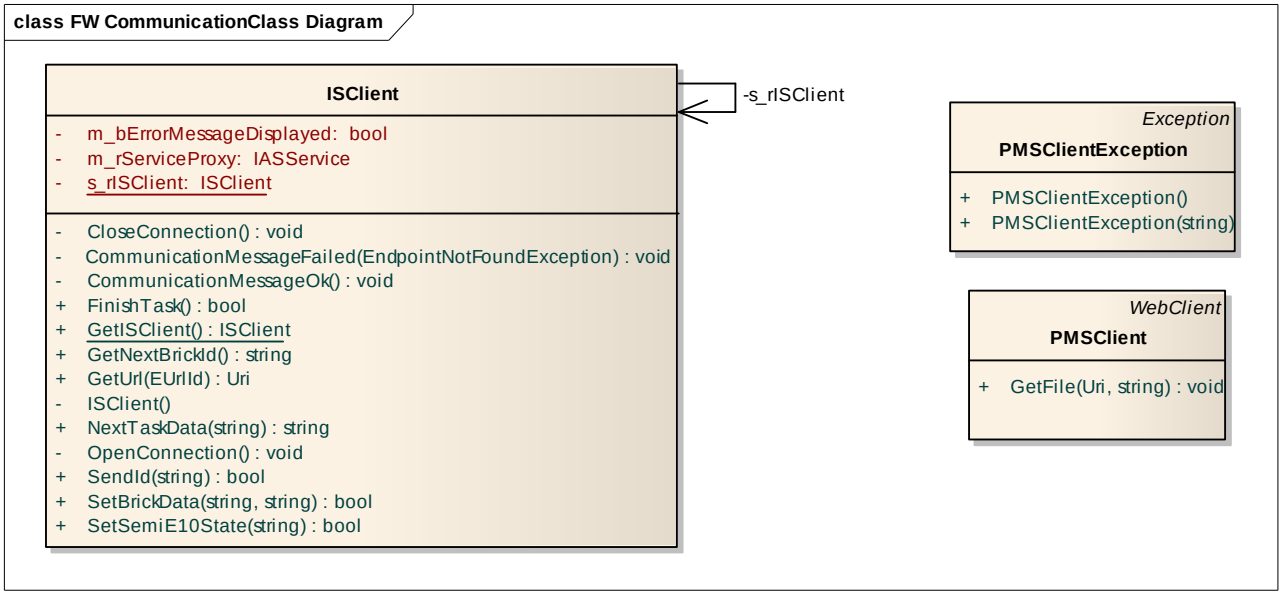


Abbildung 24: Klassendiagramm Communication

12.4.2.3 Configuration

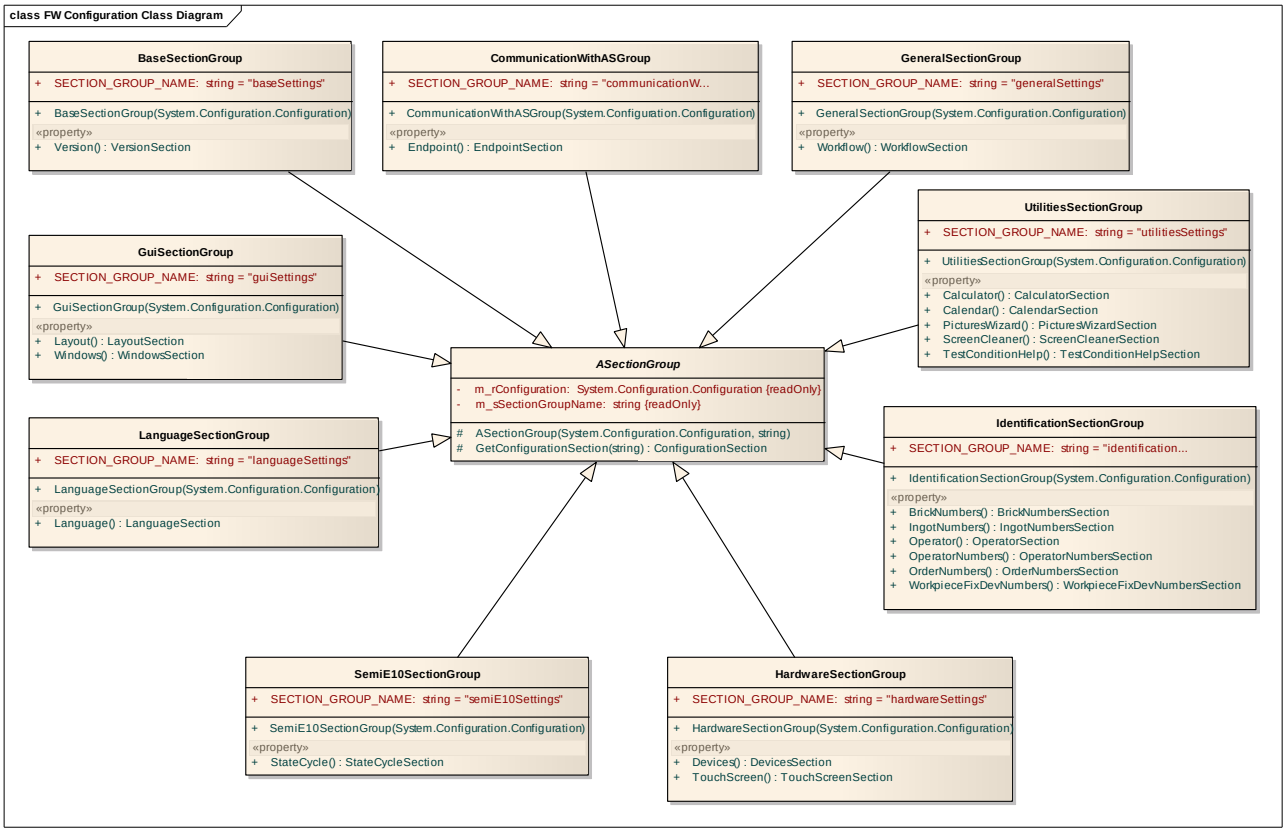


Abbildung 25: Klassendiagramm Configuration, Teil 1

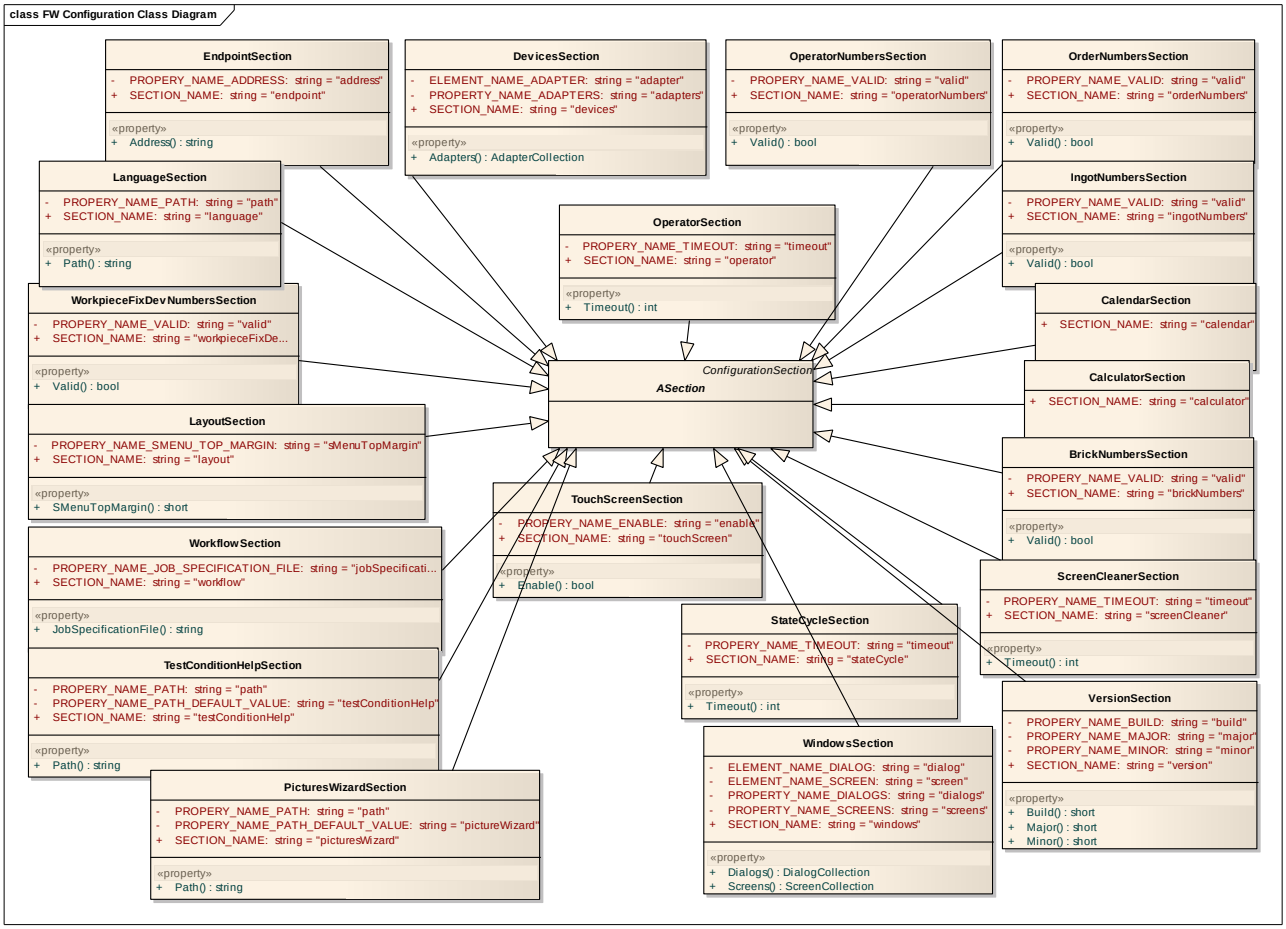


Abbildung 26: Klassendiagramm Configuration, Teil 2

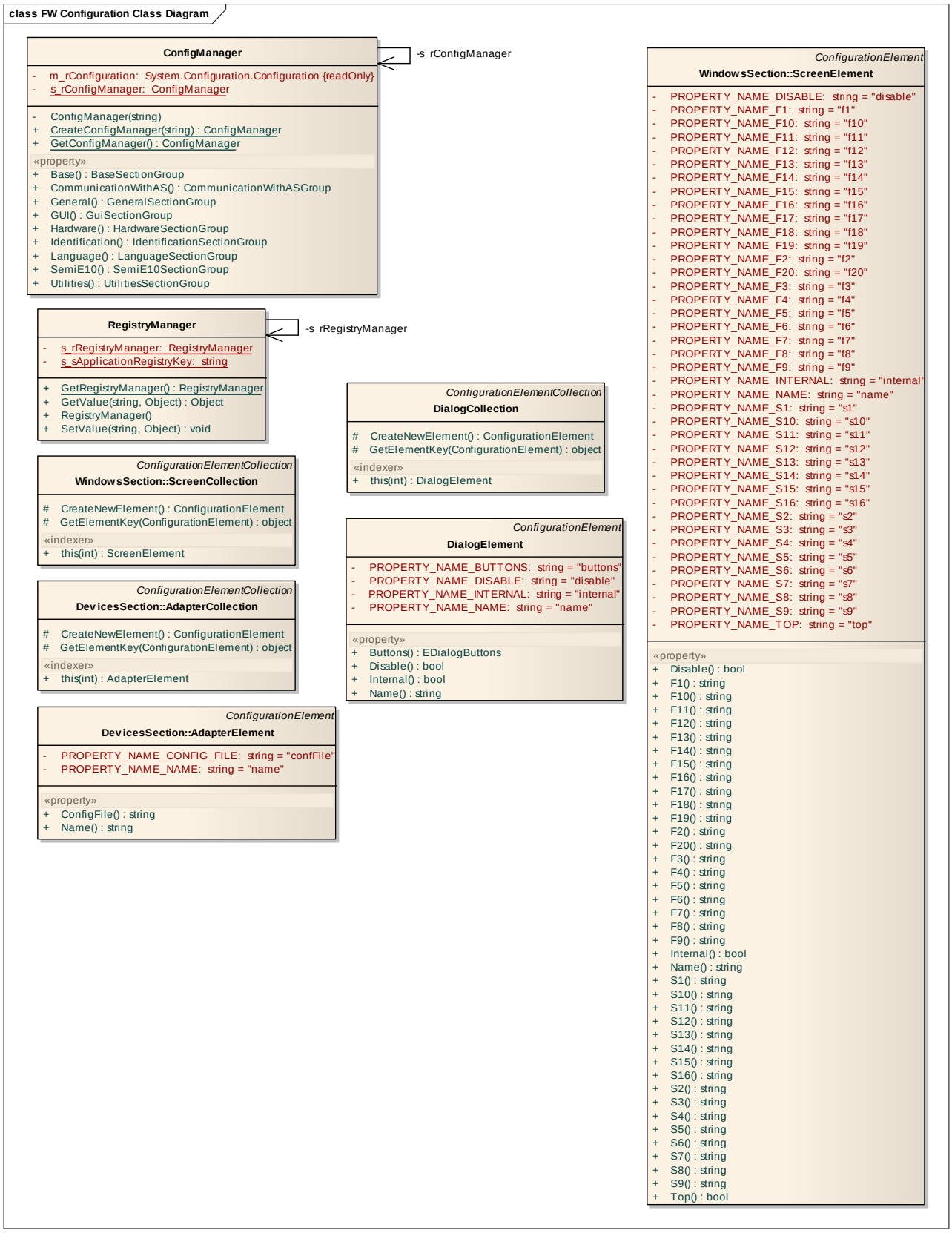


Abbildung 27: Klassendiagramm Configuration, Teil 3

12.4.2.4 GUI

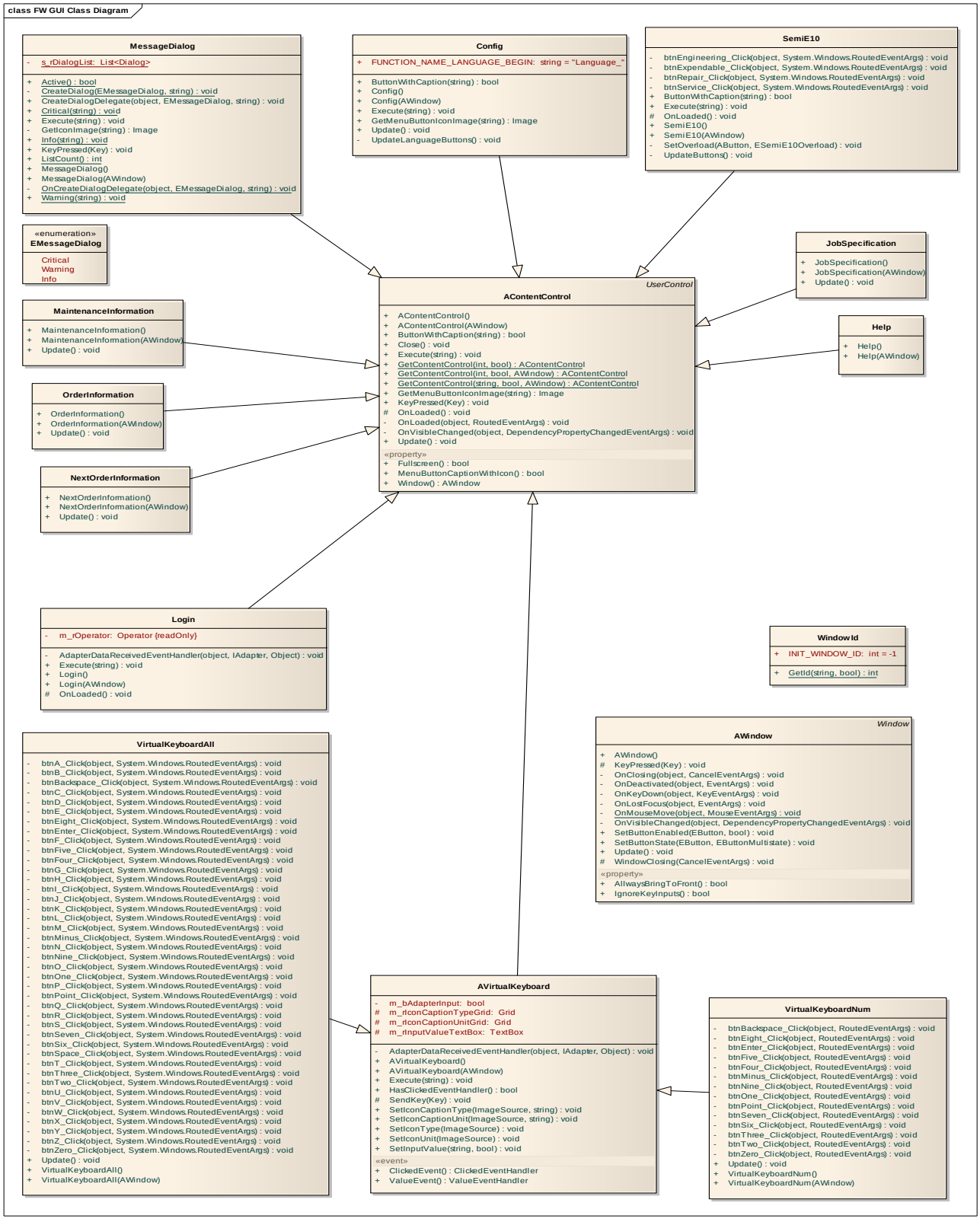


Abbildung 28: Klassendiagramm GUI

12.4.3 Referenz-Anwendung „Brick Separierer“

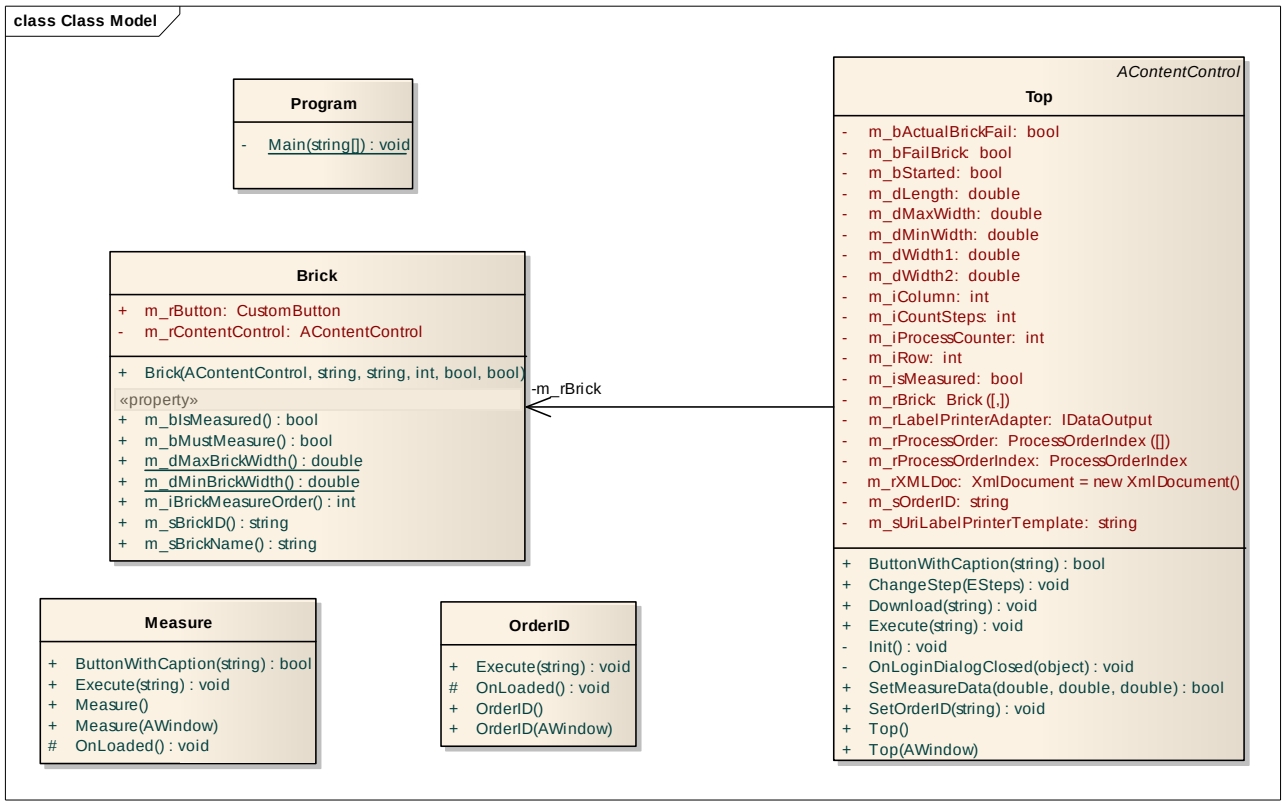


Abbildung 29: Klassendiagramm Referenz-Anwendung „Brick Separierer“